UNIVERSITY OF ALBERTA

3-D IMAGING OF REFLECTION SEISMIC WAVEFIELDS USING PARALLEL COMPUTERS

by

SOTIRIS S. KAPOTAS

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE

OF DOCTOR OF PHILOSOPHY

IN

GEOPHYSICS

DEPARTMENT OF PHYSICS

EDMONTON, ALBERTA

Fall 1991

National Library of Canada
Bibliothèque nationale du Canada

Canadian Theses Service
Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-70112-9

Canada

UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR	Sotiris S. Kapotas
TITLE OF THESIS	3-D IMAGING OF REFLECTION SEISMIC WAVEFIELDS USING PARALLEL COMPUTERS
DEGREE FOR WHICH THESIS WAS PRESENTED	Doctor of Philosophy
YEAR THIS DEGREE GRANTED	Fall 1991

Permission is hereby granted to THE UNIVERSITY OF ALBERTA LIBRARY to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only. The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author's written permission.

(SIGNED) .....

PERMANENT ADDRESS:
Eleuthériou Venizelou 47
Volos, Magnisias
Hellas (Greece)

DATED: August 8, 1991

Ithaka

When you set out for Ithaka
pray that your road's a long one,
full of adventure, full of discovery
Laistrigonians, Cyclops, angry Poseidon
don't be scared of them;
you won't find things like that on your way
as long as your thoughts are exalted,
as long as a rare excitement
stirs your spirit and your body.
Laistrigonians, Cyclops, Wild Poseidon
you won't encounter them
unless your soul raises them up in front of you.

Pray that your road's a long one.
May there be a summer morning when
full of gratitude, full of joy
you come into harbors seen for the first time;
may you stop at Phoenician trading centers
and buy fine things,
mother of pearl and coral, amber and ebony,
sensual perfumes of every kind,
as many sensual perfumes as you can;
may you visit numerous Egyptian cities
to fill yourself with learning from the wise.

Keep Ithaka always in mind.
Arriving there is what you're destined for.
But don't hurry the journey at all.
Better if it goes on for years
so you're old by the time you reach the island,
wealthy with all you've gained on the way,
not expecting Ithaka to make you rich.
Ithaka gave you the marvelous journey.
Without her you wouldn't have set out.
She hasn't anything else to give.

And if you find her poor, Ithaka won't have fooled you.
Wise as you'll have become, and so experienced,
you'll have understood by then what an Ithaka means.

Kosta Kavafys

UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled 3-D IMAGING OF REFLECTION SEISMIC WAVEFIELDS USING PARALLEL COMPUTERS submitted by SOTIRIS S. KAPOTAS in partial fulfillment of the requirements for the degree of DOCTOR OF PHILOSOPHY in GEOPHYSICS.

E.R. Kanasewich

Supervisor
E.R. Kanasewich (Physics)

G. Rostoker

G.Rostoker (Physics)

D. Schmitt

D. Schmitt (Physics)

J.S. Rogers

J.S. Rogers (Physics)

B. Joe

B. Joe (Computing Science)

R.J. Brown

External Examiner, R.J. Brown
(Geology & Geophysics,
University of Calgary)

Date: August 2, 1991

DEDICATION

Στην μνήμη της μάνας μου

Αγλαΐας

καί

στην οικογενειά μου

Σοφία, Νικολέττα–Ολγα και Μελίνα–Αλεξία

ABSTRACT

This research is oriented towards the examination and assessment of the dynamics of seismic imaging on massively parallel computers. By nature, most seismic problems carry an inherent parallelism in a subdivision by source, receiver, frequency or wavenumber. Applications are shown for 3-D prestack migration and finite-difference modeling.

Recent advances in computing based on a subdivision of the computational sequence into parallel components lead one to investigate the feasibility of seismic applications on a massively parallel computer. One must redefine the mathematical formalisms and modify the serial algorithms in favor of new ones in order to obtain the full benefit of parallel computers. The objective of this research is not only to demonstrate dramatic reductions in seismic data processing time with parallel computers but also to implement new methods of imaging that were completely impractical previously.

An analysis is made of seismic methods that are not feasible for testing on conventional mainframe computers. Emphasis is given to 3-D *prestack* migration techniques because of their challenging mathematical formulation in parallel processing, as well as the dynamic improvement they exhibit in imaging the subsurface with seismic reflection data. 3-D forward modeling methods have been developed and examined. Also, an Edge Detection method applied to seismic data in order to delineate the character of the images is presented.

The solution of seismic problems in exact 3-D, leading to higher degrees of accuracy in imaging, is discussed along with their future potential. Present results show that seismic exploration data processing appears to reflect the economic superiority of parallel computer architectures.

ACKNOWLEDGEMENTS

Professor E. R. Kanasewich deserves much credit for the euphoria of discussions and supervision of this work; to him I am deeply grateful. To the academic staff of the Physics Department I am grateful. I wish to acknowledge the invaluable assistance provided by Dr. Suhas Phadke throughout the course of my graduate work. My colleagues and especially M. Burianyk are greatly thanked.

I wish to thank Dr. S. Bharatha and H. Cooper from Esso Resources Canada Limited for providing the 3-D data for this research. Chevron Resources Ltd. is thanked for providing the 2-D Aims data set. I would like to acknowledge that major financial assistance for this research was obtained from AOSTRA (Alberta Oil Sands Research Authority), NSERC (Natural Sciences and Engineering Research Council) of Canada, and in the form of a teaching assistanteships by the University of Alberta. Also, I wish to mention that partial financial support for this work came from the Alberta Research Council and Myrias Research Corporation joint research venture.

I am grateful to the Computing Department of the Alberta Research Council and Dr. G. Rostoker from the Canadian Network for Space Research for allowing me to use their Myrias SPS computers.

The family support that a Ph.D. student needs is sometimes above human limitations. My family, companion in life Sophia, daughters Nicole-Olga and newly born Melina-Alexia, having to deal with me during the various phases of my graduate life, did provide the enormous support and understanding needed. This *laconic* paragraph is not enough to praise them, I will always try to express my "*ευγνωμοσύνη*" to them and make sure that they do not have to go through similar stages again.

Table of Contents

Chapter		Page
1.	INTRODUCTION	
	1.1.Seismic Data Processing	1
	1.2 Imaging of Seismic Sections - Definition and Methods	4
	1.3 Seismic Analysis and Parallel Computing	5
	1.4 Outline of the Thesis	8
2.	PARALLEL COMPUTERS AND THE WAVE EQUATION	9
	2.1 PARALLEL COMPUTER TAXONOMY AND ARCHITECTURE	9
	2.1.1 Introduction	9
	2.1.2 Terminology and Parallel Taxonomy	9
	2.1.3 Synchronous Architectures	10
	2.1.4 Asynchronous Architectures	12
	2.1.5 MIMD Paradigms	13
	2.1.6 Advantages and Limitations of Parallel Processing	15
	2.1.7 Serial Part of the Parallel Program	18
	2.1.8 Parallel Part of the Parallel Program	21
	2.2 THE LOGISTICS OF WAVE-EQUATION SOLUTIONS IN PARALLEL	24
	2.2.1 General Principles	24
	2.2.2. Decomposition and the one-dimentional wave equation	24
	2.2.3 Two-dimensional wave equation	29
	2.2.4 Finite differences in wave propagation	30
	2.2.5 Explicit solution of the acoustic wave equation	32
	2.2.6 Explicit finite difference solutions in parallel	34
	2.2.6.1 Coarse Parallelism	35
	2.2.6.2 Advanced Parallelism/Partial time parallelism	35

Chapter		Page
3.	2-D PRESTACK DEPTH MIGRATION IN PARALLEL	44
	3.1 ω-X Prestack Depth Migration	44
	3.1.1. Introduction	44
	3.1.2 The method	45
	3.1.3 Application description	47
	3.1.4 Spatial dependance in velocity	49
	3.1.5 Parallel algorithm implementation	54
	3.1.6 Performance monitoring on the Myrias SPS-2	55
	3.1.7 Analysis and Applications	60
	3.1.7.1 Difference operator response	60
	3.1.7.2 Synthetic examples	60
	3.1.7.3 Real Data example	72
	3.2 REVERSE TIME F-D PRESTACK MIGRATION	77
	3.2.1 Introduction	77
	3.2.2 Application description	77
	3.2.3 Extrapolation and imaging of acoustic waves	79
	3.2.4 Performance monitoring on SPS	80
	3.2.5 Reverse time migration impulse response	82
	3.2.6 Applications	82
4.	ONE PASS 3-D DEPTH MIGRATION WITH UNDER-RELAXATION	86
	4.1 3-D POST STACK DEPTH MIGRATION	86
	4.1.1 Introduction	86
	4.1.2 The Method	88
	4.1.3 Application description	89
	4.1.4 Lateral velocity variation	90

Chapter		Page
	4.1.5 Matrix solution procedure	94
	4.1.6 Under-relaxation	98
	4.1.7 Parallel implementation	100
	4.1.8 3-D migration response	102
	4.1.9 Applications of the 3-D postack migration	105
	4.1.9.1 Simulated 3-D tank model seismograms	105
	4.1.9.2 Imaging a 3-D zero - offset modeled data set	110
	4.2 PRESTACK 3-D DEPTH MIGRATION VIA UNDER-RELAXATION	116
	4.2.1 Introduction	116
	4.2.2 Method description	117
	4.2.3 Synthetic data application	121
	4.3 CONCLUSION	122
5.	3-D ACOUSTIC SEISMIC MODELING IN PARALLEL	127
	5.1 Introduction	127
	5.2 The 3-D acoustic wave equation and its F-D solution	128
	5.3 Dispersion and numerical stability	132
	5.4 Absorbing boundary conditions	134
	5.5 Solution and computer implementation	136
	5.5.1 Method of solution	136
	5.5.2 The source function	137
	5.5.3 Parallel implementation and performance	139
	5.6 Applications	141
	5.7 Some concluding remarks	149

Chapter		Page
6.	APPLICATION OF EDGE DETECTION TO SEISMIC IMAGES	150
	6.1 Introduction	150
	6.2 Zero-crossing of Laplacian-Gaussian Filtered Images	151
	6.3 Algorithm implementation	155
	6.4 Conclusion	156
7.	THE 3-D DEPTH MIGRATION	157
	7.1 DIRECT 3-D POSTSTACK DEPTH MIGRATION	157
	7.1.1 Introduction - The seismologist's dream	157
	7.1.2 The method	158
	7.1.3 Implicit F-D Solution of the diffraction term	161
	7.1.4 Complex sparse matrix solution and parallel implementation	165
	7.1.5 The exact 3-D 45 degree response	168
	7.1.6 3-D Zero offset modeling and migration	171
	7.2 3-D MIGRATION OF A SEISMIC DATA SET OVER A STEAM INJECTION REGION	175
	7.2.1 Introduction	175
	7.2.2 3-D processing and Analysis	176
	7.3 CONCLUSIONS	193
8.	EPILOGUE AND PROPHESIES	194
	BIBLIOGRAPHY	198
	APPENDICES	205

LIST OF TABLES

Table		Page
2.1	Parallel computer taxonomy	11
2.2	Asynchronous / synchronous architectures	11
2.3	Speed up for the n problem	20
2.4	Speed up for the $log_2 n$ problem	20
3.1	Performance monitoring ω-x migration	56
3.2	Performance monitoring coarse F-D migration	81
3.3	Performance monitoring advanced F-D migration	81
5.1	Performance monitoring 3-D acoustic modeling	140
7.1	Performance monitoring exact 3-D ω-x-y migration	166

LIST OF FIGURES

Figure		Page
2.1.	Myrias SPS architecture.	16
2.2.	Schematic representation of parallelism in a code.	17
2.3	Parallel decomposition of the code.	23
2.4	Parallel concurrency in terms of wall building: (a) overlapping topology, (b) pipeline, (c) irregular and (d) triangular (pyramid).	27
2.5.	The vibrating string with a typical decomposition onto a concurrent computer.	28
2.6	Coarse parallelism scheme.	36
2.7.	(a) Finite difference solution in the x-t domain using a propagating difference star; (b) same approach using sub-domains in the x direction resulting in a triangular time stepping.	38
2.8.	Schematic representation of the 2-D pyramidal space (domain of dependence).	39
3.1.	Pseudocode of the omega-x migration as implemented in parallel.	57
3.2.	Graphical representation of the omega - x migration.	58
3.3.	Time vs PE's for omega - x migration.	59
3.4	Speed up over 64 PE's.	59
3.5	(a) Desired impulse response for migration, (b) 15^0 approximation, (c) 45^0 approximation and (d) 65^0 approximation.	61
3.6	Step fault model used to test the prestack migration method.	62
3.7	Generated synthetic shot gather for step fault model via ray tracing.	63
3.8	Prestack depth - migrated shot record for step fault model.	64

Figure		Page
3.9	Final prestack depth migrated stack of the fault model. The character of the fault is clear, justifying the accuracy of the method.	65
3.10	Geological model used to generate synthetic seismic records.	67
3.11	A few representative examples of synthetic shot records for the model in Figure 3.10.	68
3.12	Prestack depth migrated shot gathers from Figure 3.11.	69
3.13	Common Depth Point stack for the model in Figure 3.10.	70
3.14	Prestack depth migrated stack of the model in Figure 3.10.	71
3.15	Location Map showing the seismic line in Southern Alberta.	74
3.16	CDP stacked section of a portion of the line from the Princess well area.(from Kanasewich and Phadke, 1988)	74
3.17	Shot record (a) and prestack migrated counterpart (b).	75
3.18	Prestack depth migrated stack of the Princess Well Data and Interpreted section (from Kanasewich and Phadke, 1988)	76
3.19.	Impulse responses of fourth order reverse time migration.	83
3.20	Prestack reverse time migrated shots.	84
3.21	Prestack reverse time migrated stack of the step fault model.	85
4.1	Pseudocode of the 3-D under-relaxed migration	101
4.2	(a) Desired impulse response for migration, (b) 15^0 3-D, under-relaxed (c) 15^0 3-D exact response.	103
4.3	(a) Depth slice of the 3-D under-relaxed migration response, (b) the 15 degree direct 3-D equivalent.	104
4.4	Diagrammatic sketch of (a) planar and (b) cross sectional views of the 3-D tank model	106

Figure		Page
4.5	2-D seismic section of line 35 from the tank model experiment.	107
4.6	3-D migrated plot of line 35. The positions of the reflectors form the top of the reef and fault (817 m), the first interface at (945 m) and the base of the model (1100 m) have been imaged correctly. The sides of the reef and fault having a slope of 45^0 were not properly imaged	108
4.7	3-D migrated depth slice of the tank model (830m). The reef and fault characteristics of the model are evident.	109
4.8	Diagrammatic representation of the physical model used for zero offset modeling.	112
4.9	3-D zero offset seismic response of the model shown in Figure 4.8	113
4.10	(a) 3-D migrated sections via the under-relaxation method lines 9,10,11 (b) 3-D migrated result using the exact 3-D solution.	114
4.11	(a) 3-D migrated sections via the under-relaxation method lines 14,15,16 (b) 3-D migrated result using the exact 3-D solution.	115
4.12.	Pseudocode of the prestack 3-D under-relaxed migration	120
4.13	Schematic representation of the 3-D model used for the generation of synthetic seismograms.	123
4.14	Shot record generated via 3-D forward modeling using the model in Figure 4.13.	124
4.15	Migrated shot record of the data in Figure 4.14. Top cross sectional view, bottom planar views at 450m and 500m.	125
4.16	Prestack migrated stack section of the 3-D step fault model.Depth slices shown are at depths of 450m and 500m., located at the top of the reflectors present in the model.	126

Figure		Page
5.1	Gaussian source function for f_0=30Hz and Δt=1ms.	138
5.2	A perspective plot showing the 3-D step fault model used for modeling	143
5.3	A synthetic 3-D shot record generated over the step fault model.	144
5.4	A vertical view of y=41 from the shot in Figure 5.3. The diffractions from the fault edge are evident.	145
5.5	A vertical view of x=20, 41 and 60 from the shot in Figure 5.4. Diffractions from the sides of the fault are present.	146
5.6	A horizontal view at t=500ms from the shot in Figure 5.4. The reflection character of the fault has been modeled properly.	147
5.7	Snapshot for t=200ms. X-Y, X-Z and Y-Z planes showing the propagation character of the wavefield in 3-D media.	148
6.1	Parameters related to the LoG operator	153
6.2	Response of a step to a LoG mask.	154
7.1	Exact 3-D 45 degree pseudocode.	167
7.2	(a) Desired impulse response for migration, (b) 45^0 3-D direct, (c) 45^0 3-D onepass	169
7.3	Depth slices at 200m for the response of (a) 45^0 3-D direct, (b) 45^0 3-D onepass.	170
7.4	3-D zero offset response for lines 9,10,11,14,15 and 16.	172
7.5	3-D migrated results for lines 8,9,10,14,15 and 16 respectively.	173
7.6	2-D planar view of the edges of the syncline. Direct 3-D migration (a) onepass 3-D migration under-relaxed	174
7.7	Plan view of the survey in Cold Lake.	177

Figure		Page
7.8	Cross section of Common Midpoint stacks for lines 44,45 and 46 before (a) and after (b) steam injection.	178
7.9	Cross section of the velocity-depth model used for 3-D migration.	179
7.10	3-D migrated cross sections of lines 44,45 and 46 before (a) and after (b) steam injection.	180
7.11	Planar view of depth slices at 300m and 600m migrated via the exact 3-D solution.(a) before steam injection (b) after injection.	184
7.12	Steam injection well locations drawn to scale with the 2-D images from the seismic data for comparison.	185
7.13	2-D depth slices at depths 476 to 488m migrated via the exact 3-D solution.(a) before steam injection (b) after injection.	186
7.14	2-D depth slices at depths 476 to 488m migrated via the onepass 3-D with under-relaxation solution.	187
7.15	2-D depth slices of Figure 6.13 after the SVD enhancement (a)before steam injection (b) after injection.	188
7.16	2-D depth slices of Figure 7.14 after the SVD enhancement (a)before steam injection (b) after injection.	189
7.17	2-D depth slices of Figure 7.15 after Edge detection (a)before steam injection (b) after injection	190
7.18	2-D depth slices of Figure 7.16 after Edge detection (a)before steam injection (b) after injection.	191
7.19	Comparison of a depth slice (500m) between (a) exact 3-D migration and (b) onepass 3-D with under-relaxation in order to evaluate the accuracy of the later.	192

CHAPTER 1

INTRODUCTION

1.1 Seismic Data Processing

The seismic reflection method was invented in 1917 (Fessenden) and developed for geophysical prospecting in the 1930s (DeGolyer, 1935; McDermott, 1932; Weatherby, 1940) as a single channel analog recording and processing technique. Essentially it is an echo technique, similar to radar, making use of artificially induced elastic waves. The seismic waves are reflected from subsurface impedance changes, usually due to rapid changes in compressional wave velocity due to variations in geologic stratigraphy. In the early years a single analog signal, or trace, was recorded and, after the application of rather crude analog filtering operations, analyzed visually by a geophysicist for information content relating to the presence or absence of hydrocarbon traps. Later, methods were developed for utilizing the information content of multiple traces through special processing techniques to improve the signal - to - noise ratio of the recorded signals.

In the late 1950s and early 1960s, digital recording and computer processing of seismic traces was begun. Extremely large volumes of data were collected in the field with very little spatial or temporal filtering so that optional signal enhancement techniques could be applied in the data processing centers. A significant advantage of a large amount of redundant information is that the power of the seismic source (explosion or mechanical vibration) could be made small with little impact on the physical environment.

The emphasis on large volumes of data distinguishes seismic processing from other petroleum supercomputer applications. Seismic processing possesses the potential for system imbalance (I/O vs. CPU) in low *computational complexity processing* techniques.

Advanced processing techniques represent a requirement for 10^4 times the number of computations represented by the largest presently practical algorithms, but these advanced techniques require even greater volumes of data as the sample rates and number of recorded channels increase.

A typical data acquisition scenario today is the recording of 6 s of data at a sample rate of 2 ms for 48 channels per shot (source excitation) with 15 shots per km. This represents 2.16 million samples per km, or equivalently 8.64 megabytes per km for 32 - bit samples. Increasingly common, however, is the recording of 0.5 ms data on 1000 channels or more with 100 or more shots per km. With increased dynamic range and three - component geophones, a 6 s recording will produce 1.2×10^9 samples or 4.8 Gigabytes per km.

The end product of most seismic processing sequences is a seismic section, which is a graphical display of the subsurface geological structures and their physical properties. Seismic sections often present the wave form due to ground displacement, velocity sensitivity or pressure as a function of time and spatial location. Increasingly, though, seismic sections are displayed in color to represent a larger dynamic range of amplitudes or some computed properties of the crust. These could be the compressional or shear wave velocities, the Poisson's ratio, or changes in reflection coefficients with angle of incidence which indicate variations of porosity or type of fluid present.

Although somewhat oversimplified, it will be adequate to classify seismic processing into two major categories: *conventional* and *advanced.*

Conventional seismic processing has evolved over the years from the earliest single channel methods and to a great extent, is a trial-and-error method to enhance the cosmetic appearance of the cross section. Mathematically, the processing techniques are similar to other signal processing applications used in sonar signal processing. These are stacking, spatial and temporal filtering (convolution) to reduce noise, and deconvolution to increase the resolution. The convolution operation (represented in the time-domain by the

accumulation of inner or dot products) is by far the most common operation.

In the last 20 years, frequency domain techniques employing the Fast Fourier Transform have become increasingly popular. The *stacked* section is the usual end product of conventional seismic processing. It is the result of stacking or adding together multiple traces which, after an operation known as normal moveout has been performed, represent the estimated responses at the same point in space. Normal moveout is the Pythagorean operation based on an assumption (more or less invalid depending on the geology) of flat horizontal reflecting surfaces and an estimated velocity profile of the subsurface. In general, conventional processing assumes simple geometrical ray paths and linear convolutional operators.

Advanced processing techniques are distinguished primarily by their use of the wave equation, which introduces an element of mathematical rigor and computational complexity not found in conventional techniques. The most common advanced processing technique is migration, which can be thought of as an imaging technique similar to that used in optics.

Migration is performed in various combinations of time, frequency, space and wavenumber domains. Migration can be performed either before or after the stacking operation, but prestack migration is so computationally intensive that its use has only very recently become feasible with the present generation of supercomputers. Two-dimensional migration is much faster than three-dimensional, but is less accurate due to its assumption of a two-dimensional world. As a computational compromise, a pseudo three-dimensional migration technique known as two-step or tandem three-dimensional migration is often used.

While migration is an inverse procedure, a forward procedure known as forward modeling is used to produce a synthetic seismic section. Most production forward modeling programs are based on ray tracing, although finite difference and finite element

methods based on a scalar or vector form of the elastic wave equation exist and these overcome some of the limitations of ray tracing.

Most migration and forward modeling procedures currently in use are based on a simplified form of the wave equation known as the scalar or acoustic wave equation. What is desired is the use of the full elastic wave equation in which the spatial distribution of compressional and shear wave velocities can be derived. The use of the elastic wave equation is growing but is limited by inadequate computational power and the expense of acquiring three component field data.

Generally speaking, seismic data are amenable to parallel processing. This is particularly true for conventional processing where individual traces are processed as independent channels. Parallel computation in the course of advanced seismic processing involves numerical analytic issues such as matrix factorization, the numerical solutions of ordinary differential equations, Fourier and Hilbert transforms, and the computation of eigenvalues and eigenvectors of matrices via singular value decomposition. It is the scope of this work to investigate these issues of parallelism for those advanced processing techniques, defined here within the context of seismic imaging.

1.2 Imaging of Seismic Sections - Definition and Methods

Accurate methods for migration, modeling and image processing of seismic records, that anatomically examine the intrinsic information needed for geophysical interpretation and exploration purposes are developed in this thesis. The term *imaging of seismic sections* is used within the context of this thesis to indicate those methods and techniques which reveal information contained in seismic observations. This study concentrates on the solution of those geophysical problems, such as prestack seismic migration in both two and three dimensions, which make use of massively parallel computers.

In order to be able to attain the most reliable interpretation of the available seismic data

a geophysicist should apply a minimum number of assumptions. In order to achieve this goal a step must be taken beyond the conventional seismic analysis. In this thesis I will show that in order to properly image seismic data we need to solve the exact 3-D problem with a limited number of assumptions and that it is necessary in practice to utilize the power of available supercomputers to achieve this goal.

As described in the previous section, advanced seismic methods based on mathematical rigor and computational complexity include seismic migration and forward modeling. The choice of these two methods along with image processing will comprise the set of applications developed here to analyze seismograms with the aid of parallel computers. The methods used include: prestack migration in two dimensions using the principle of double downward continuation in both frequency-space and time-space domains; a new development of the one-pass 3-D poststack and prestack migration; image processing of seismic sections based on 2-D transforms such as singular value decomposition and edge detection. 3-D acoustic forward modeling is used to simulate data to be used by the 3-D migration method. Finally, direct 3-D solutions of the migration problem are presented in the poststack mode.

1.3 Seismic Analysis and Parallel Computing

A new phenomenon witnessed during the end of the 80's was the realization of the dynamic potential of parallel computing. Serial computations based on single CPU systems are approaching physical barriers based on the speed of electrons and systems based on photons have not been realized in a practical format as yet. In any case they will still be limited by the speed of light. The new *parallel* computers offer the user the ability to execute the program more quickly and run larger applications. Parallelism is also a natural phenomenon. Our brain is a good example of parallelism in nature. It can deal

with several informational parameters in order to achieve a single task; (e.g.) vision and hearing input can be used to make conclusions about an action to be taken. Similarly our everyday living world is parallel in that several workers can complete a job faster than a single worker could. The operation of multi-stage tasks sequentially could be attributed to either computer limitations or human physical boundaries. In this thesis it is demonstrated that this type of technology allows one to image seismic reflection data in ways that have not been possible previously.

The field of supercomputing has during the last decade seen the development of a group of computers known as vector computers which also have some parallel potential. Vector computers allow operations to be carried out simultaneously at high speeds, in a sequential order. Recently, the so called parallel computers have been introduced which can at the same time execute independent parts of a program. In vector computers the programmer must look at ways of programming long sets of floating point operations. In a similar way parallel computers require the user to break up the program into discrete independent parts, each with a significant amount of work.

The primary use of such supercomputers is known to occur in seismic exploration and in the oil industry in general. These computers allow seismologists to improve their computational speed as well as the resolution of results. At a resent symposium on Computers in Geophysics the following conclusion was drawn,

> "..nevertheless, because nearly all geophysical problems display an inherent parallelism and because it's the only way on the horizon to achieve improvements of several orders of magnitude in computer performance, geophysical programs will eventually come to be written in true parallel form under parallel operating systems."
>
> (Leading Edge, March, 1990).

This type of emphasis in parallel computing in seismology runs in conjunction with our recent investigations in this field.

This research will focus primarily on the newer class of parallel computers, especially the type of MIMD (Multiple Instruction Multiple Data) architectures, which contain moderate (32) to massive (> 200) numbers of processors, as potential seismic processing and imaging systems. A variety of such computer topologies have surfaced. Current research among a range of applications has shown that promising levels of speed can be achieved by this type of computational environment.

Limited by the availability of only one class of MIMD, the new massively parallel computer Myrias SPS-2 and SPS-3 (Scalable Parallel Supercomputer), this work is oriented towards the examination and assessment of the dynamics of seismic analysis on this type of computer topology. The logistics of the solutions and algorithm development can be easily extended to other parallel computers with minimal effort. Applications are shown to those advanced seismic methods mentioned above. The degree of parallelism possible in the above algorithms and also for those of finite-difference solutions will be discussed. Present - day results have shown that seismic exploration data processing, because of the multiplicity of sources and receivers and the decomposition by Fourier transform into a large number of monochromatic plane waves, is well suited scientifically and economically to reflect the economic superiority of parallel architectures.

1.4 Outline of the Thesis

Chapter 1 contains an overview of the background and direction of this research work.

In Chapter 2 the taxonomy of the present - day parallel computers is examined along with their characteristic potentials and a generalized analysis of wave equation solutions is presented for parallel systems.

An elegant method for 2-D prestack depth migration in parallel will be presented in Chapter 3 in the frequency-space and time-space domains, followed by synthetic and real examples.

Chapter 4 will cover the subject of one-pass 3-D depth migration, based on the principle of the Alternating Direction Implicit (ADI) and a new *under-relaxation* approach for the solution of the 3-D vector differential equation. Applications to synthetic data and real data recorded over an Enhanced Oil Recovery Area (EOR) will be shown.

Chapter 5 covers the forward modeling solution of the acoustic wave equation in 3-D as implemented in a parallel computer and programmed to simulate seismic responses.

Chapter 6 is a description of the Edge Detection method developed for application to Seismic Sections.

In Chapter 7 the solution of the exact 3-D seismic depth migration problem will be shown and implementations in parallel along with applications will be discussed.

Chapter 8 is a brief overview summarizing the original sections presented in this thesis. In addition some prophecies are made.

CHAPTER 2

PARALLEL COMPUTERS AND THE WAVE EQUATION

2.1 PARALLEL COMPUTER TAXONOMY AND ARCHITECTURE

2.1.1 Introduction

The diversity of parallel computer architectures can bewilder the scientist in today's rapidly expanding supercomputing world. The discussion in this thesis will not be directed towards an analysis of those architectures but primarily to the advantages and limitations of parallel processing viewed from the eyes of a geophysical user. Recent advances in parallel processing technology are based on hardware systems such as systolic, hypercube and interconnection. An examination of these terminologies will be given below, along with a technical review of parallel computer hardware.

2.1.2 Terminology and Parallel Taxonomy

It has become increasingly difficult to arrange parallel computers into a categorical pattern because of the many variations that are available commercially. It is useful to keep the logical taxonomy of Flynn (1966) in mind and possibly augment or modify it accordingly. Flynn's taxonomy is based on the intercorrelation of instructions and data which yields four basic categories:

SISD (single instruction - single data)

-serial computers.

MISD (multiple instruction - single data)

-impractical design to use.

SIMD (single instruction - multiple data)

-multiple processors simultaneously executing the same instruction on different data.

MIMD (multiple instruction - multiple data)

-multiple processors autonomously executing diverse instructions on diverse data.

In order for someone today to generalize the definition of these architectures, a modified Flynn scheme must be used. Duncan (1990) proposed one that includes appropriate computers that Flynn did not foresee and excluded architectures with low-level parallelism. His basic taxonomy was separated into synchronous and asynchronous systems with an additional category of MIMD paradigms. Table 2.1 shows this taxonomical layout and Table 2.2 diagrammatically shows the difference between synchronous and asynchronous architectures.

2.1.3 Synchronous Architectures

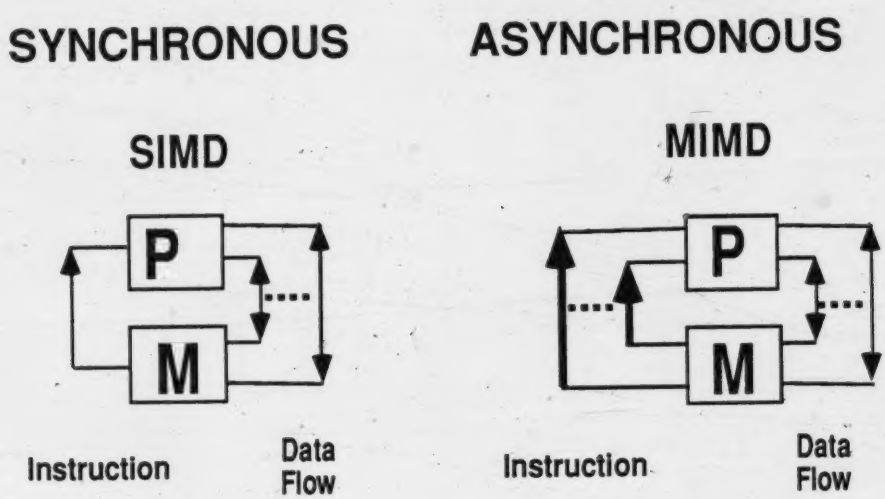
Pipelined vector processors.

This type of computer supports massive matrix and vector operations. Some vector processors are characterized by multiple pipelined functional units (Hwang, 1984), which can operate concurrently. This type of system provides parallel vector processing by sequentially streaming vector elements into a functional unit pipeline.

TABLE 2.1 Parallel computer taxonomy

	SYNCHRONOUS	
Vector	SIMD	Systolic
	ASYNCHRONOUS	
	MIMD	
Distributed memory		Shared Memory
	MIMD Paradigm	
MIMD/SIMD	Dataflow	Reduction
Wavefront	Myrias SPS	

TABLE 2.2 Asynchronous / synchronous architectures



SIMD (Single Instruction - Multiple Data)

This type of environment employs a central control network for processor-to-processor or processor-to-memory communication. The network allows instruction results calculated at one processor to be communicated to another processor for use. This kind of structured architecture is used for large scale scientific calculations, such as image processing, nuclear energy modeling, seismic modeling and others. The main drawback of such systems is the complicated programming environment.

Systolic.

This configuration is used to solve special purpose problems, such as signal processing. It includes pipelined multiprocessors in which data is pushed in rhythmic fashion through the network of processors before returning to memory.

2.1.4 Asynchronous architectures (MIMD)

Here multiple processors can execute independent instruction streams, using local data. MIMD computers run mostly in an asynchronous manner, requiring each processor to work independently. This type of architecture supports a higher level of parallelism (subprogram and task levels). The cost effectiveness of such systems encourages MIMD experimentations. The two basic categories of those environments are *distributed* and *shared* memory.

Distributed memory systems

Processing nodes are connected with a processor to a processor intercommunication network. Nodes slave data by passing messages through the interconnection network. This fairly new platform has been constructed to provide a scalable multiprocessor architecture

that will satisfy the needs of large scale scientific calculations characterized by local data references. Several topologies exist of this form but will not be mentioned here as they are of different scope to this research.

Shared memory architectures

Interprocessor coordination is accomplished in this design by providing a global, shared memory that each processor can access. Shared memory computers might not have the problem of message passing that distributed memory architectures have, but they have difficulty to retain a synchronized data access manner.

2.1.5 MIMD Paradigms

An extension of MIMD computers by several architectures makes possible the introduction of this taxonomy. Each one of these is predicated on MIMD principles of asynchronous operation and concurrent manipulations of multiple instructions with data streams.

MIMD/SIMD architectures

A number of designs during the 80's allowed for selected portions of MIMD architecture to be controlled in SIMD fashion. This type of flexibility allows for further future research with the intention to support parallel image processing and expert system applications (Stolfo and Miranker, 1986).

Dataflow architectures

The power of the dataflow paradigm lies in its potential to enable instruction for execution as soon as all their operands become available. In case of data dependencies

dataflow will exploit concurrency at task, routine and instruction levels.

Reduction architectures

Reduction architectures execute programs that consist of nested expressions, which are aimed to produce a result; thus the entire program (reduction) is ultimately reduced to its result. In this case an instruction is enabled for execution when the results are required as operands for another instruction already enabled for execution. Practical challenge to this architecture is to include synchronized demands.

Wavefront arrays

The processors of this configuration employ systolic data pipelining - data is pulsed in rhythmic fashion from memory to processors - with an asynchronous dataflow execution paradigm. This architecture is characterized by modular processors and regular, local interconnection networks, such that greater scalability can be attained. This configuration permits simpler programming.

The Myrias SPS Parallel Computer

Myrias SPS is a massively parallel computer. It has been designed to expand to *n* processors, with the largest system assembled to date involving 1044 processors. Although it has some features similar to the Ncube, the processors are not custom-designed, but rather off the shelf. This design feature likely compromises performance but allowed the company to incorporate new chips quickly as they become commercially available. It also allowed the computer to ride the steep performance gain curve associated with the chips. A very important feature with the Myrias computer is that the operating system handles all data motion, program synchronization, distribution and merging. There are only a couple of extensions to standard FORTRAN 77 or C, the principal extension

being a parallel DO loop (PARDO) (Myrias, PAMS, 1990). This affords an enormous advantage to the programmer and allows for porting software to this computer easily. Figure 2.1 shows the architecture of the SPS. Although it was only commercially available for approximately a year, there is already an large range of applications running on this parallel computer (Stone et al. 1990; Kapotas, et al. 1990, 1991).

2.1.6 Advantages and Limitations of Parallel Processing

In this section, a very brief introduction to some of the basic issues involved in parallel processing will be presented. The following discussion is aimed at the reader who is unfamiliar with the subject of parallel computing. Figure 2.2 shows a simple schematic representation of a computer program drawn as a large square block to the left of the figure. The program is composed of subroutines denoted by rabbits and a turtle. For the sake of argument most of these subroutines might be small and could be independently executed in parallel. The "turtle" subroutine might be large and can only be calculated slowly in series. If the program is to be executed in a serial computer the sequence followed is shown in the upper right part of the scheme. Some of the smaller subroutines could be executed first followed by the larger one followed by the rest of the smaller ones. If a parallel computer is used as shown in the middle portion of the figure, the small independent subroutines could be executed simultaneously, followed by a *bottle neck* serial executable subroutine, completing the run with a final parallel execution sequence.

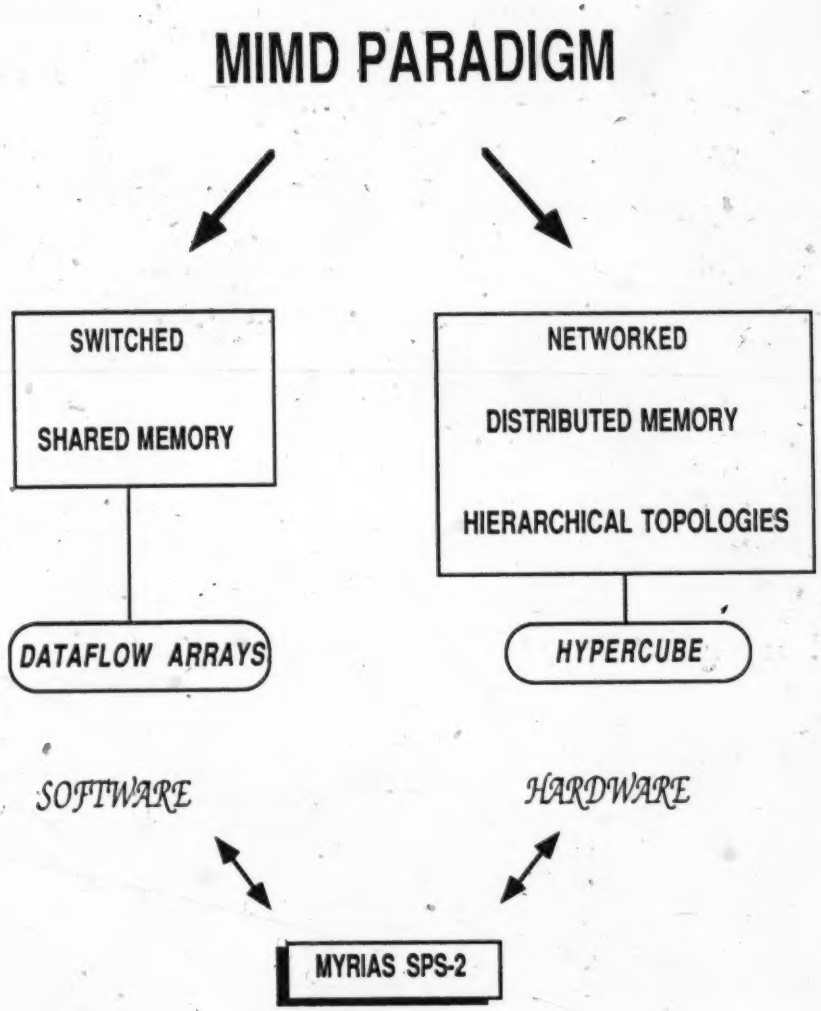


Figure 2.1. Myrias SPS architecture.

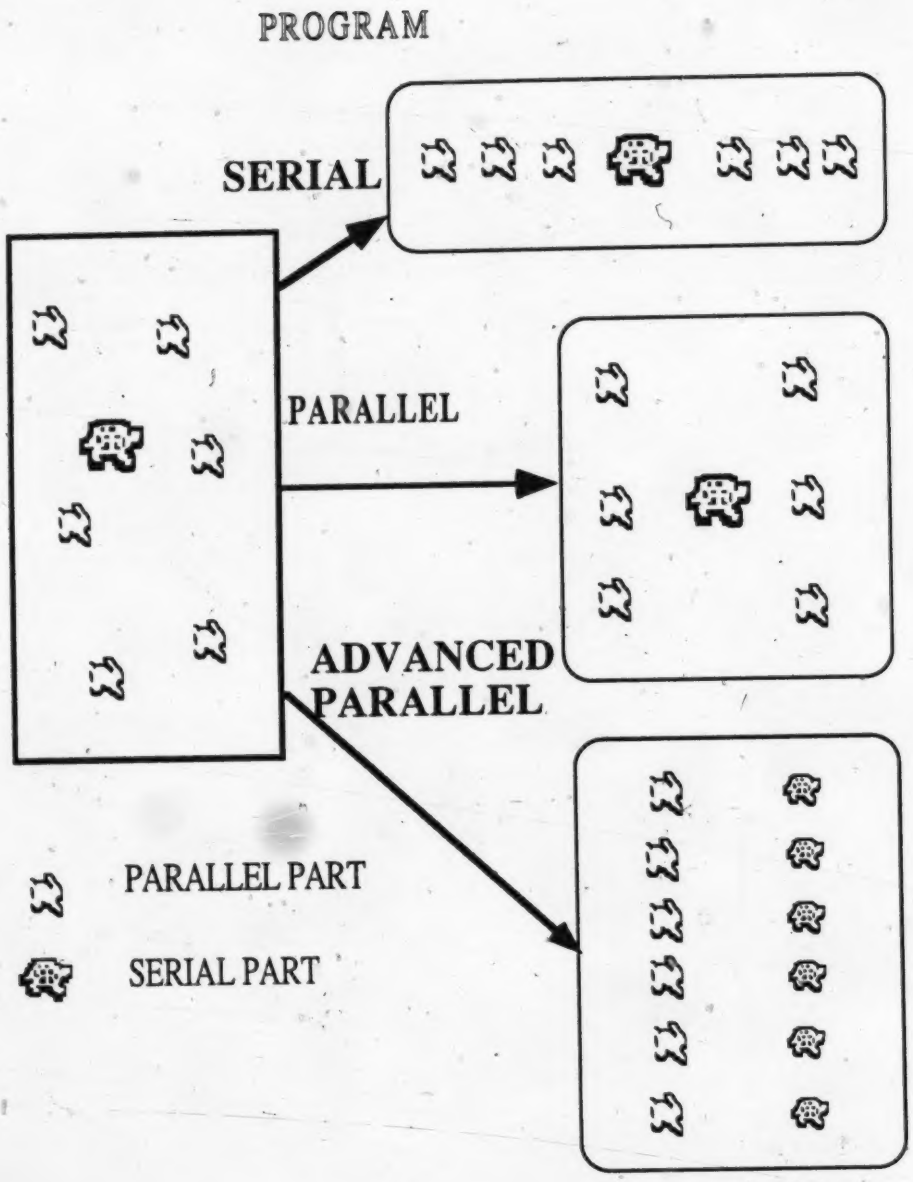


Figure 2.2. Schematic representation of parallelism in a code.

One usually does not stop at this point of coarse parallelism level but takes the program to a further stage. In such a case one has to go back to the original mathematical equation of the problem and discover higher degree of parallelism by making an algebraic formulation. The lower right part of Figure 2.2 shows a possible way of resolving parallelism. This reduces the long serial routine in a way so that it could be executed in parallel along with each of the other parallel subroutines. It is clear that the ability to run parts of this program in parallel will result in a saving in total execution time.

2.1.7 Serial part of the parallel program

To simplify the logistics of parallel programming let us examine behavior of serial and parallel parts of a program from the view point of parallel programmability. Returning to Figure 2.2, it can be seen that if the program is executed on a serial computer, the larger subroutine has an important effect on execution time. When the program is run on a parallel computer the larger subroutine has a significant effect on the overall computational time. This will be examined in the following paragraphs. It is assumed that the reader is familiar with the FORTRAN language, which is commonly used in scientific applications.

In Table 2.3 we consider a program skeleton consisting of a DO loop (part of a program that uses changing index i for calculation). The DO 10 loop, whose index i ranges from 1 to n and which ends with a 10 CONTINUE statement, involves calculating the vector $\boldsymbol{a}$. Each member of a associated with each index i takes, for the shake of argument, one second of CPU (central processor unit) time. Suppose that a parallel computer was available that enabled each calculation associated with each index i to be executed simultaneously on different processors; hence the word *parallel* in brackets before the DO statement. Following the DO loop is a portion of the program that must be executed sequentially. For purposes of illustration, suppose that as n becomes larger the

time taken to calculate this portion is proportional to n/10 seconds of CPU time.

To execute this program on a serial computer would take $n + n/10$ seconds of time (one second of time for each index i which ranges from 1 to n giving n seconds for the loop plus $n/10$ seconds for the remainder of the time), whereas a parallel computer would need $1 + n/10$ seconds (one second for the loop because the calculation for each index i is executed simultaneously, plus $n/10$ seconds for the remainder). The *speedup* or relative computational advantage, which is defined as the serial time divided by the parallel time, is equal to $(n + n/10)/(1+n/10)$. As n becomes larger and approaches infinity, this function approaches 11.

Now consider, as in Table 2.4, a second example with a similar program execution. The DO loop is identical to that shown in Table 2.3 and described above. For each index i which ranges from 1 to n, a member of a vector a is calculated; this calculation takes one second of computer time for each i. The only difference in this example is that the serial portion following the DO loop takes $log_2 n$ (as for FFT's)seconds of CPU time rather than $n/10$, as above. To execute this program, the serial time is now $n + \log_2 n$ seconds, whereas a parallel computer would take $1 + log_2 n$ seconds. Speedup is $(n + log_2 n) / (1 + log_2 n)$ which approaches n as n approaches infinity. For large n, this is a dramatic improvement in speedup over the above case, in which the speedup was at most 11.

As an illustration in these examples, if $n=1000$, using 1000 processors in which the time to execute the serial portion of the program varied as $n/10$, the speedup in Table 2.3 is 10.9. In the second example (Table 2.4) in which the time to execute the serial program varied as $log_2 n$, the speedup is 250. This dramatic difference is strictly due to the time spent executing the serial portion of the program.

TABLE 2.3 Speed up for the *n* problem

```
      (Parallel) DO 10 i=1,n
          a(i) = something that takes one second of cpu (i) time
10    continue

          Something that takes n/10 seconds of cpu seconds
```

Serial time = n + n/10 seconds

Parallel time = 1 + n/10 seconds . . [for n processors]

Speedup = Serial time / Parallel time = (n +n/10)/(1+n/10) ~ 11 as n goes to infinity

---

TABLE 2.4 Speed up for the $log_2 n$ problem

```
      (Parallel) DO 10 i=1,n
          a(i) = something that takes one cpu (i) second
10    continue

          Something that takes log2n cpu seconds
```

Serial time = n + $\log_2 n$ seconds

Parallel time = 1 + $\log_2 n$ seconds

Speedup = Serial time / Parallel time = (n +$\log_2 n$)/(1+$\log_2 n$) ~= n as n goes to infinity

---

2.1.8 Parallel part of the parallel program

Figure 2.3 again shows the original program composed of a number of small subroutines and one large one. Suppose a parallel computer that allowed all three of the small subroutines to be executed in parallel was available. As an alternative, consider a parallel computer that allowed only two of these subroutines to be executed synchronously. On the second computer, two of the subroutines would be executed in parallel, then the third would be executed, followed by the remainder of the program. The second computer would take a longer time to finish executing the program because it lacks sufficient processors to compute all the possible parallel tasks together. This leads to an issue called scalability which is important to the massively parallel variety of computers.

Scalability relates a computer's performance to its size. A system is termed *scalable* if an increase in system size (e.g. number of processors) produces an analogous increase in its computational power (speed) (Fox, 1989). Peak rate is an often misused term that gives a theoretical performance rate for a particular machine. This rate is unachievable on all but the most particular of problems but gives a measure of a computer's power. A parallel system with a small number of processors, say one to eight, is said to have low scalability. Medium to high scalability may describe a computer with the number of processors ranging from 25 to hundreds. If an application program with potential for parallel processing contains many more parallel tasks than it has available processors, then a computer's ability to increase its functional size and to execute those tasks defines its scalability.

2.1.9 Programmability of parallel computers.

The advent of vector computers, such as any Cray or the CDC 205, allows the

programmer to achieve speedups when the program is changed to process many multiplication and addition operations sequentially. Similarly, with today's newer variety of parallel computers the ability of the programmer to keep a large number of available processors working is critical to achieving high performance (Eisner, 1989). If the computer's compiler can achieve this automatically it is easier for the programmer to obtain fast execution times. Otherwise the computer is said to have low programmability. Programmability may be defined as the ability to easily achieve good program performance and the term is applicable to any computer being manufactured. It is known that, on some of the currently available parallel machines, high performance can be achieved only through extensive reformulation of the algorithm or through extensive use of a machine language whereas, in others of this class parallelism can be easily achieved using a high level language with few programming changes.

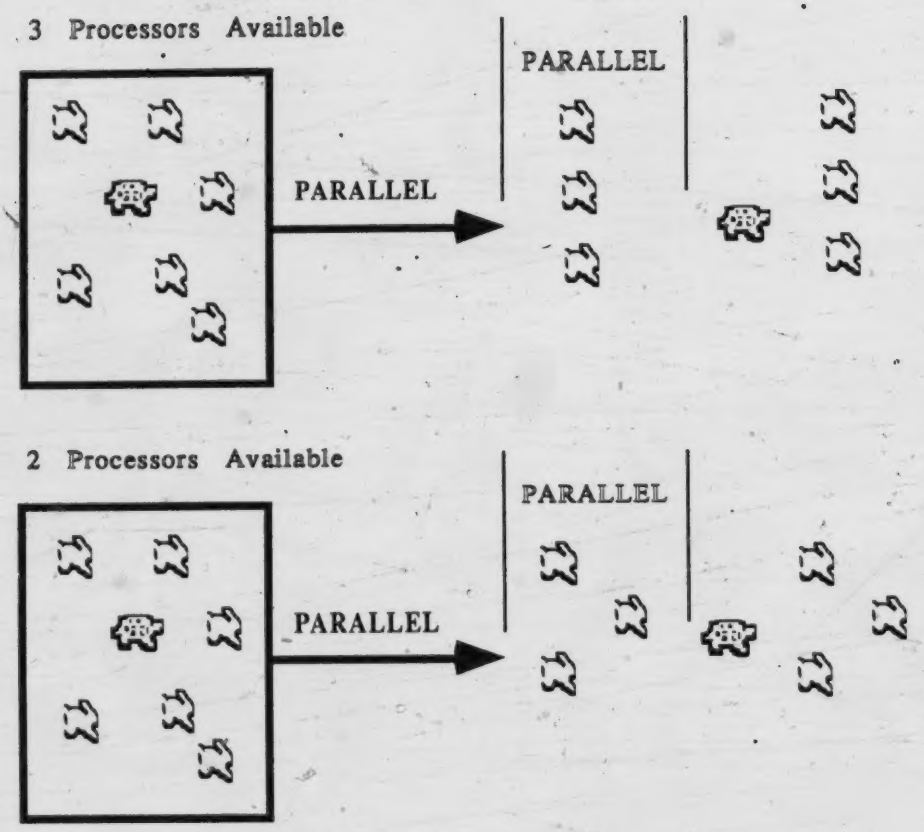


Figure 2.3. Parallel decomposition of the code.

2.2 THE LOGISTICS OF WAVE-EQUATION SOLUTIONS IN PARALLEL

2.2.1 General principles

The purpose of the section is to introduce some of the methods used to design efficient parallel algorithms constructed to solve an important class of equations. The solution of the wave equation as used by seismologists will be discussed and its transformation for programming on the Myrias SPS will be shown. The principal assumptions are similar to those used in the introduction to describe the effect of concurrency in the achievement of several tasks. A useful analogy is the example of bricklayers trying to build a wall as seen in Figure 2.4.

The simplest way to view the wave equation is to look at the one dimensional case which describes the motion of a string in one dimension. We will show an initial concept for such system solutions and then extend the application to the solution of the two dimensional acoustic (or full acoustic) wave equation in two domains, time and frequency. Both domain solutions are important in seismology because they closely describe elastic wave phenomena during processes of modeling, imaging or inversion.

This section will be devoted to the application of parallelism in the solution of these systems of equations in either one or two dimensions.

2.2.2 Decomposition and the one-dimensional wave equation

In order to obtain a better understanding of the decomposition of wave equation solutions in terms of concurrency I will compare this to the problem of constructing a wall by a team of masons. As we proceed, the relation of this problem to wave equation solutions will become evident. Some principles of domain decompositions have been

discussed by Fox et al. (1989), for a specific type of parallel computers. Here an extension is shown of the methods to account for the architecture of the SPS parallel computer.

Figure 2.4 shows several possibilities in building a wall from the point of view of decomposition. Topologies of overlapping geometries, pipelining, gapped and triangular shapes are shown. The domain is composed of many members and most decomposition assigns large numbers of members to each node. In the case of building the wall, the domain is the wall itself and the members are the individual bricks. In the case of the wave equation, the domain is the physical extent of the string and the displacement $\Psi(x,t)$ is a function of distance, x, and time, t, representing the unknown solution (Figure 2.5) driven by the equation:

$$\frac{\partial^2 \Psi}{\partial x^2} = \frac{1}{c^2} \frac{\partial^2 \Psi}{\partial t^2}$$

Let us now look at the analogy of decomposition of the building of the wall to that of the wave equation describing the motion of the string. The upper part of Figure 2.4 (a) shows an overlapping approach to wall building. The concurrency lies in the decomposition of the work into equal regions with some overlapping in between, so that each mason completes that particular region assigned to him passing only common bricks at the overlapping stage. This derivation is based on vertical segmentation. This particular scheme could be used for the solution of the wave equation of the string by decomposition as shown in Figure 2.5. In this case each processor has been attributed a limited number of points for clarity. If one should need to improve efficiency then one must allocate a larger number of points to each processor. The connection topology shown in the figure requires that each processor communicate with its nearest neighbor in the one dimensional chain. This is a subset of the connections available in a hypercube type parallel computer.

Figure 2.4(b) shows a schematic representation of pipelining tasks. In this case the masons work together to build the wall decomposed into horizontal sections. An extension of this method will be discussed later on in part (d) of the figure. Solving the wave equation in this domain will require partial completions of each time step and task communication from processor-to processor. This type of solution might be easier to apply on a SIMD type of parallel computer but its efficiency will be very low.

Case (c) of irregular geometry usually requires some advance parallel techniques in order to retain load balancing. Without appropriate computer architecture this might be difficult to achieve. Allocation of tasks has to be carefully chosen and the amount of work per task equally weighted. The dynamic load balancing available in the SPS series of parallel computers is a necessary factor for efficiently solving these type of problems.

As an alternative to a time domain solution one has the triangular *pyramidal* method as shown in Figure 4(d). By realizing that each point's value at time *t* is only related to its neighbor's value at time *t-1* we could in fact have a partial parallelization over time. This scheme requires a large number of nodal points on a specific grid in order to operate efficiently. This approach will be described in some detail during the analysis of the 2-D wave equation solutions and the reverse time finite-difference migration application.

This section has compared time domain solutions of the wave equation to those of wall building. However, in seismic applications a transformation and solution of the wave equation in the Fourier domain is widely used. This type of transformation allows for a different type of approach in order to investigate parallelism on an application. The logistics for parallel computations in the frequency domain will be described in the framework of the 2-D case.

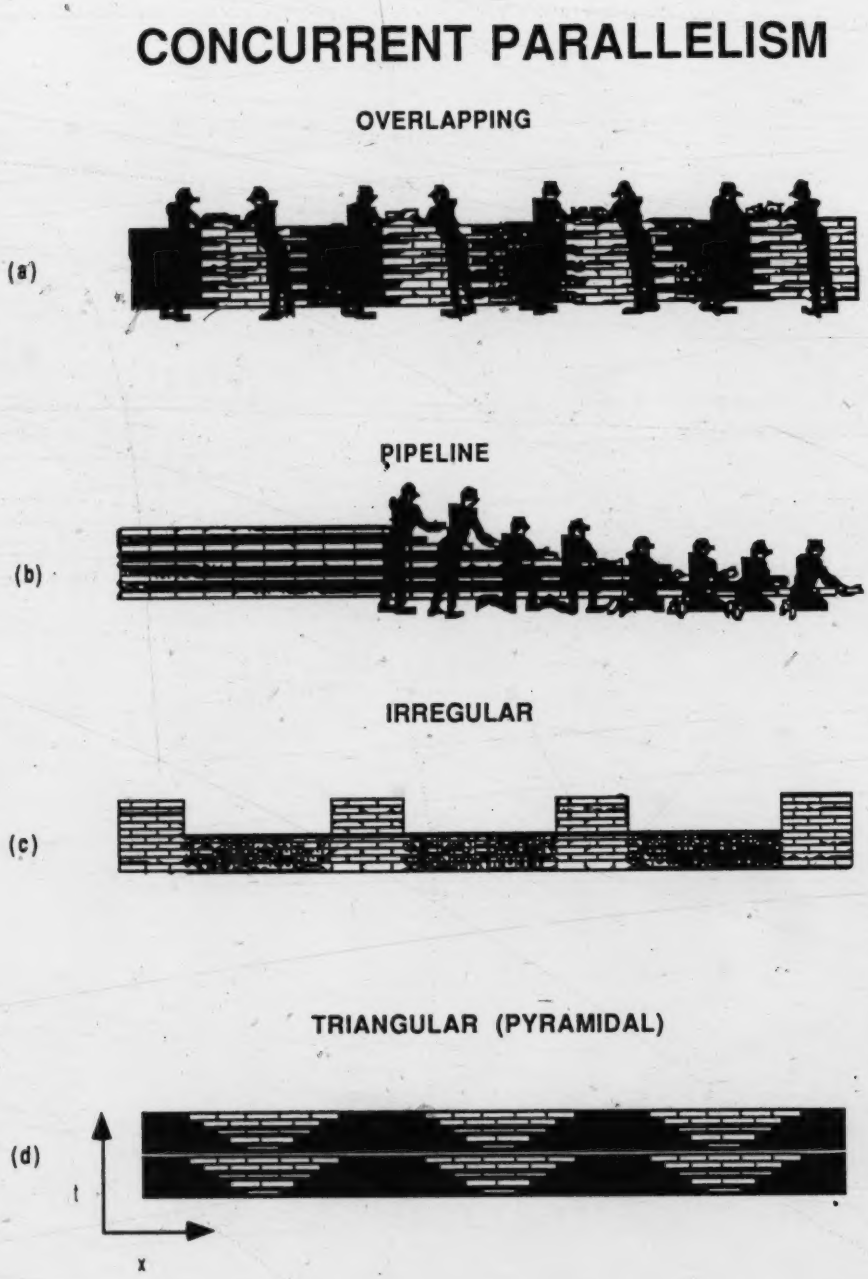


Figure 2.4. Parallel concurrency in terms of wall building: (a) overlapping topology, (b) pipeline, (c) irregular and (d) triangular (pyramid).

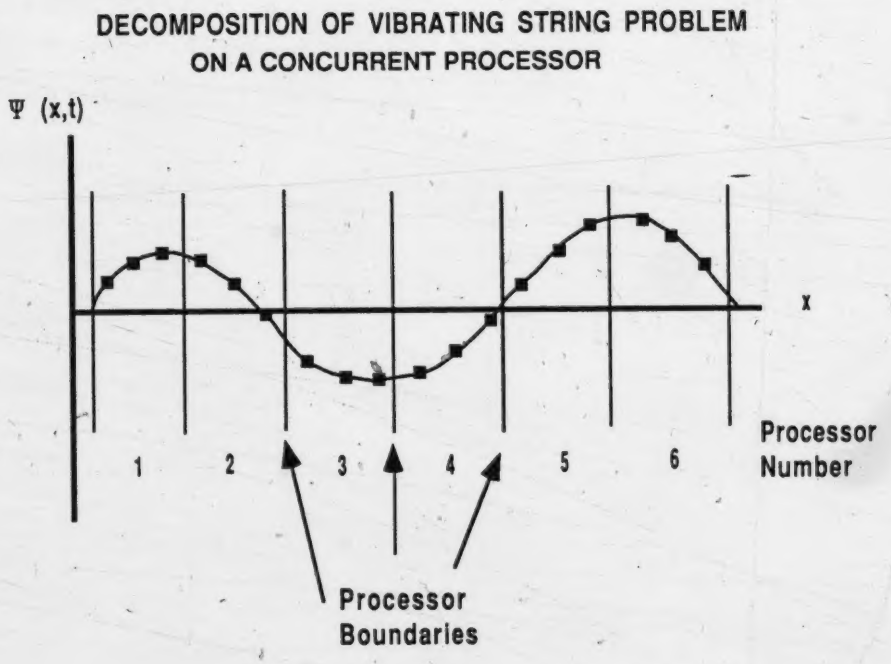


Figure 2.5. The vibrating string with a typical decomposition onto a concurrent computer. Each node contains 3 grid points at which Ψ is defined.

2.2.3 Two-dimensional wave equation

The 2-D acoustic wave equation describes sound waves in a liquid or a gas and is widely used in seismic applications. Another set of equations in vector form describe elastic waves in solids and will not be discussed here. In reflection seismology the elastic or acoustic waves generated by small near-surface sources are reflected from subsurface discontinuities in density and elastic moduli and are recorded on a large number of discrete surface receivers, usually arranged linearly. The subsurface discontinuities are imaged by extrapolating the wavefield of the echoes to the reflecting interfaces. This process is called migration. The wavefield extrapolation used in seismic migration is an approximation in which one assumes that it is possible to use a two dimensional full acoustic wave equation of the form:

$$\frac{\partial}{\partial x}\frac{1}{\rho}\frac{\partial}{\partial x}P(x,z,t)+\frac{\partial}{\partial z}\frac{1}{\rho}\frac{\partial}{\partial z}P(x,z,t)=\frac{1}{K}\frac{\partial^2}{\partial t^2}P(x,z,t) \qquad (2.1)$$

where $P(x,z,t)$ is the pressure of the observed wavefield. Because density only varies by a few percent one also often assumes that it is not a function of x and z. The seismic velocity, $v(x,z)$ or the modulus of incompressibility, K, can vary by at least one order of magnitude. Then equation (2.1) can be reduced to the following scalar wave equation:

$$\frac{\partial^2}{\partial x^2}P(x,z,t)+\frac{\partial^2}{\partial z^2}P(x,z,t)=\frac{1}{v^2(x,z)}\frac{\partial^2}{\partial t^2}P(x,z,t) \qquad (2.2)$$

where $v^2(x,z) = K/\rho$.

It would be more precise to use this type of equation for seismic migration purposes, but only a limited number of applied industrial migration algorithms use this form. One of

these, reverse - time migration will be described later in this thesis. Most migration techniques use what is called a paraxial (or parabolic) approximation to this equation. We will first investigate this type of equation and describe the parallel solution implementation in terms of wavefield extrapolation and then I will discuss time domain solutions of the scalar wave equation using finite-difference approximations. The derivation of the parabolic wave equation is shown in Appendix A.

Parallelism.

Rewriting $P(k_x,z,\omega)$ in equation (A8) in terms of matrix form for only four spatially Fourier transformed field values one has:

$$\begin{bmatrix} P_{k_1} & P_{k_2} \\ P_{k_3} & P_{k_4} \end{bmatrix}_{\omega_n}^{z+1} = exp\left(-i\int_0^z k_z(z)dz\right)\begin{bmatrix} P_{k_1} & P_{k_2} \\ P_{k_3} & P_{k_4} \end{bmatrix}_{\omega_n}^{z} \qquad (2.3)$$

The main parallel part of this application relies on the fact that each temporal frequency component P_{ω_n} can be calculated simultaneously for all depths. Therefore, the algorithm can be constructed to allocate tasks to each processor and solve all depth extrapolations at each frequency term. The option of applying parallelism along the spatial frequency is also possible. This formulation has not been tested on the SPS MIMD computer but it can be done easily. An alternative formulation allows one to approximate for lateral (spatial) velocity variations. This formulation is used in the migration applications examined in this thesis and will be shown in separate chapters.

2.2.4 Finite differences in wave propagation. Time domain solutions.

Let us now look back at the hyperbolic scalar wave equation:

$$\frac{\partial^2}{\partial x^2} P(x,z,t) + \frac{\partial^2}{\partial z^2} P(x,z,t) = \frac{1}{v^2(x,z)} \frac{\partial^2}{\partial t^2} P(x,z,t) \qquad (2.4)$$

Finite difference approximations to this differential equation give rise to difference equations. In finite difference solutions, both spatial and time variables are discretized by superimposing a rectangular grid on the model.

Kelly et al. (1976) pointed out two basic formulations for the solution of this type of equation: The *homogeneous* approach, which solves the wave equation imposing boundary conditions explicitly on the interfaces between layers, and the *heterogeneous* approach, which solves directly the equation for the whole model. While these concepts relate to the wave equation solutions in the forward domain, we are interested in this case for the inverse domain solutions. The approach is basically similar.

Finite difference schemes are classified into two categories: explicit and implicit. In explicit schemes the response is evaluated at an advanced time exclusively from the response already determined by the previous times. In implicit modes the response is evaluated simultaneously at all spatial locations at an advanced time from known values at previous times.

Various difference schemes exist in the literature applied to hyperbolic systems of equations (Clifton, 1967; Santosa and Pao; 1986, Mitchell, 1969). One must choose an appropriate scheme based on the equation used and the computational power available. Sun and McMechan (1986) applied the explicit second-order difference code for their application to reverse time migration. Similarly Fricke (1988) used the same approach to solve the problem on a SIMD (Thinking Machines) parallel computer.

Our interest in this section is directed towards finite-difference solutions of the wave equation in parallel. Discussions of the implementation of these solutions to reverse - time migration will be shown in Chapter 3.

It seems that explicit schemes are more popular in the solution of seismic wave propagation because of their straightforward formulation. Care should be taken in the choice of spatial interval to reduce memory requirements as well as avoiding aliasing. Also the choice of acoustic velocities should be such that the stability condition is satisfied. First we will give an explicit scheme (second order) for the solution of the equation above, followed by implementation methods for a parallel processor.

2.2.5 Explicit solution of the acoustic wave equation

Equation (2.4) requires some difference approximations for the second partial derivatives in order that it may be implemented in a computer. The most obvious approach is to use derivative definitions . A first derivative might be approximated as:

$$\frac{\partial P}{\partial t} \approx \frac{P_{t+1} - P_t}{\Delta t} \tag{2.5}$$

The second derivative formula in terms of time can be obtained by taking the first derivative twice. This gives:

$$\frac{\partial^2 P}{\partial t^2} \approx \frac{P_{t+1} - 2P_t + P_{t-1}}{\Delta t^2} \tag{2.6}$$

Similarly for the spatial components x:

$$\frac{\partial^2 P}{\partial x^2} \approx \frac{P^{x+1} - 2P^x + P^{x-1}}{\Delta x^2} \tag{2.7}$$

and z:

$$\frac{\partial^2 P}{\partial z^2} \approx \frac{P^{z+1} - 2P^z + P^{z-1}}{\Delta z^2} \qquad (2.8)$$

Substituting (2.6), (2.7) and (2.8) into (2.4) we obtain:

$$\frac{1}{v^2}\frac{P_{t+1} - 2P_t + P_{t-1}}{\Delta t^2} = \frac{P^{x+1} - 2P^x + P^{x-1}}{\Delta x^2} + \frac{P^{z+1} - 2P^z + P^{z-1}}{\Delta z^2} \qquad (2.9)$$

Choosing $\Delta x = \Delta z$ and separating variables of the advanced time step we have:

$$P_{t+1}^{x,z} = 2P_t^{x,z} - P_{t-1}^{x,z} + \frac{v^2\Delta t^2}{\Delta x^2}\left[P_t^{x+1,z} + P_t^{x-1,z} + P_t^{x,z+1} + P_t^{x,z-1} - 4P_t^{x,z}\right] \qquad (2.10)$$

Equation (2.10) gives the wavefield at time $t+1$ from values of the field at times t and $t-1$ at all grid points x, z.

In order for this scheme to be stable, the following condition must hold true (Alford et al,1974):

$$q = \frac{v^2\Delta t^2}{\Delta x^2} \leq \frac{1}{2} \qquad (2.11)$$

If this condition is satisfied the equation above in its homogeneous form can be used to extrapolate the wavefield in the reverse time order (Chapter 3).

For the sake of illustration we only derived the second - order difference scheme approximation to the scalar wave equation. In a similar manner one can obtain a fourth order differencing scheme for the same equation. Equation (2.10) has been programmed on a parallel MIMD computer (Myrias SPS) using two different approaches. One uses a coarse parallelism and the other has partial parallelism over time.

2.2.6 Explicit finite - difference solutions in parallel

The calculation of finite differences can be computationally expensive when the spatial resolution is high and/or the velocities of the model are quite large. The efficiency of the method depends greatly on the technique of calculating the wavefield amplitudes at each time step.

The advent of parallel computing opens new avenues in the applications of finite differences. In this section a new concept of parallel operations will be described corresponding to the two-dimensional solution of equation (2.10). Here a description of algorithm formulation in terms of coarse parallelism over space is shown for the second order difference scheme for illustration purposes. The fourth order scheme can been analysed similarly.

Equation (2.10) can be written in a matrix form in order that parallelism in the method will become apparent. Let us consider four time advanced values of $P(x,z,t)$ for a fixed horizontal location x; then the pressure would be $P(z,t)$, for illustrative purposes.

This yields:

$$\begin{bmatrix} P^1 \\ P^2 \\ P^3 \\ P^4 \end{bmatrix}^{t+1} = \begin{bmatrix} 2\text{-}2q & -q & 0 & 0 \\ -q & 2\text{-}2q & -q & 0 \\ 0 & -q & 2\text{-}2q & -q \\ 0 & 0 & -q & 2\text{-}2q \end{bmatrix} \begin{bmatrix} P^1 \\ P^2 \\ P^3 \\ P^4 \end{bmatrix}^{t} - \begin{bmatrix} P^1 \\ P^2 \\ P^3 \\ P^4 \end{bmatrix}^{t-1} \qquad (2.12)$$

Examining the second order matrix formulation of the acoustic wave equation solution we can see an important parallel aspect for each time step. All spatial grid points at time t_{i+1} are calculated independently and require only information from time t_i and t_{i-1} using a set of simultaneous equations.

2.2.6.1 Coarse parallelism

Based on the observations above one can introduce parallel task allocation into each grid point location. The ultimate goal would be to have as many processors as grid points. But the amount of calculation at each grid point is not that floating point intensive to benefit from the capabilities of a MIMD computer. If one is using processors capable of doing a large number of floating point operations then one must assign tasks with an equivalent number of operations (at all x positions) at each time step.

This first assumption for coarse parallelism was used to program the equation above in parallel. Parallelism runs over one of the spatial coordinates (z in our case), allowing the other spatial domain to be calculated serially inside each task for higher efficiency, thus obtaining the time advanced 2-D grid points all simultaneously. This application is of great benefit to the solution of problems of this type. The faster a task is completed the smaller the time becomes between temporal steps. One, of course, has to deal with overhead problems in this cases (closing and opening of parallel tasks) but if the size of the problem is large the overhead can be reduced. Figure 2.6. shows how this type of parallelism is used for reverse time migration.

2.2.6.2 Advanced parallelism / partial time parallelism

Usually in a finite difference solution of a physical problem in the time domain one seeks geometrical relationships between grid points that might reduce the computational times. In order to understand the geometry present in a matrix formulation of the problem, for equation (2.12), one examines the behavior of the difference operator over time.

Figure 2.7 (a) displays a two dimensional grid showing the propagation of a difference star along x and t. At each advanced time step information from three previous space

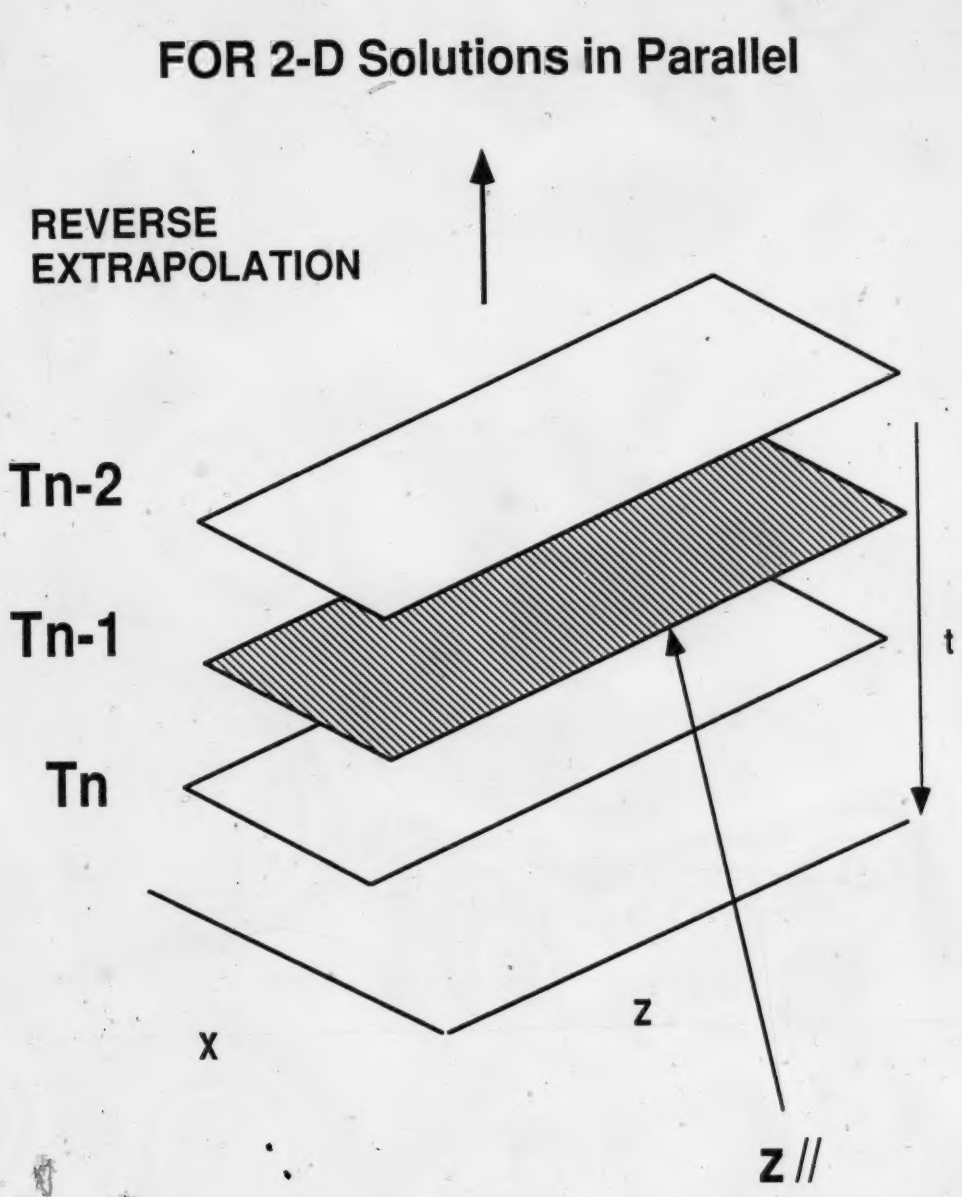


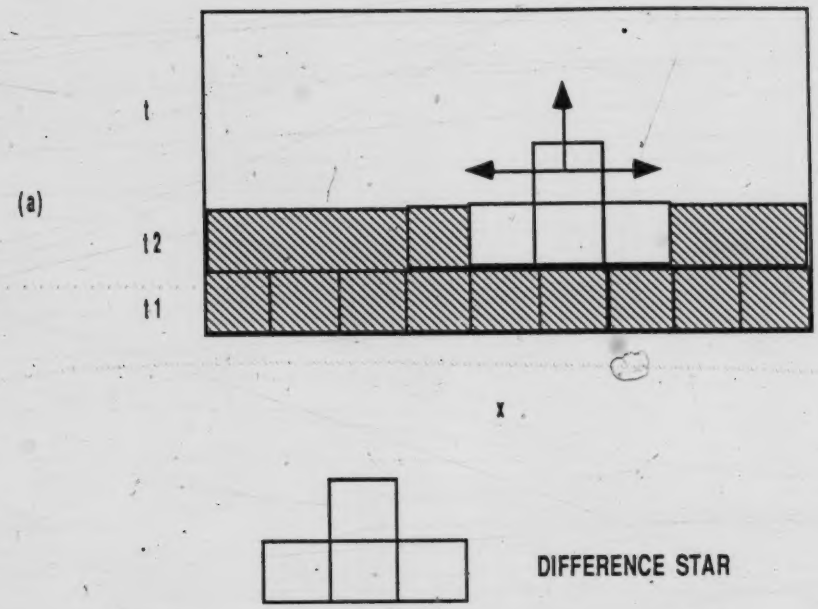
Figure 2.6 Coarse parallelism Scheme.

points is necessary. As time progresses this starlike operator moves along yielding estimates for the required grid point. If one wants to subdivide the x direction into separate domains (Figure 2.4(d), section 2.2.2, Figure 2.7(b)), as in the case of wall building, the propagating star has to be constrained at a specific sub-domain and eventually it will converge to a point. Basically this process creates a triangle like propagation geometry for the updated solution. Similar triangles have been constructed for the other sub-domains. Once the convergence point has been reached, because no other information from neighbors is available, a reverse order star operator (triangle builder) fills in the troughs towards the convergence level. Then the processes continues to the final time step.

This type of symmetrical dynamics for the difference operator allows us to consider the existence of partial parallelism over time. In fact, one can design an algorithm that assigns sub-domain tasks (length N) that will converge after (N/2 -1) time steps, and then reverse the order of task allocation to calculate the gap values.

The idea for this computational process came up initially during discussions with Myrias personnel (J. Foster, personal communication). In this research an initial implementation of this type of approach has been carried out and tested on a reverse time migration code.

The algorithm basically works in 2-D space where the geometrical representation of the triangle is a pyramid. The process, in fact, builds pyramids to an analogous time level by allocating parallel tasks numbered over squared block divisions. Reverse building then takes place to fill in information. Care must be taken on the implementation of the source function that drives the difference field, or the input data in the case of migration. Figure 2.8. shows a diagrammatic representation of the method in 3-D, as the build-up of the pyramid accumulates.



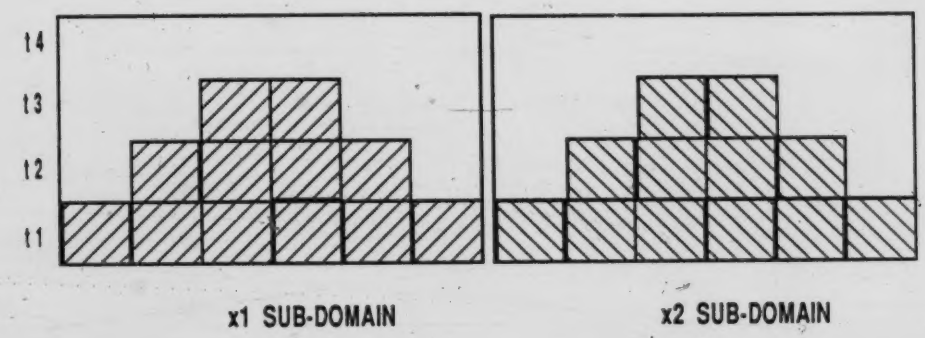


Figure 2.7. (a) finite difference solution in x-t domain using a propagating difference star; (b) same approach using sub-domains in the x direction resulting in a triangular time stepping.

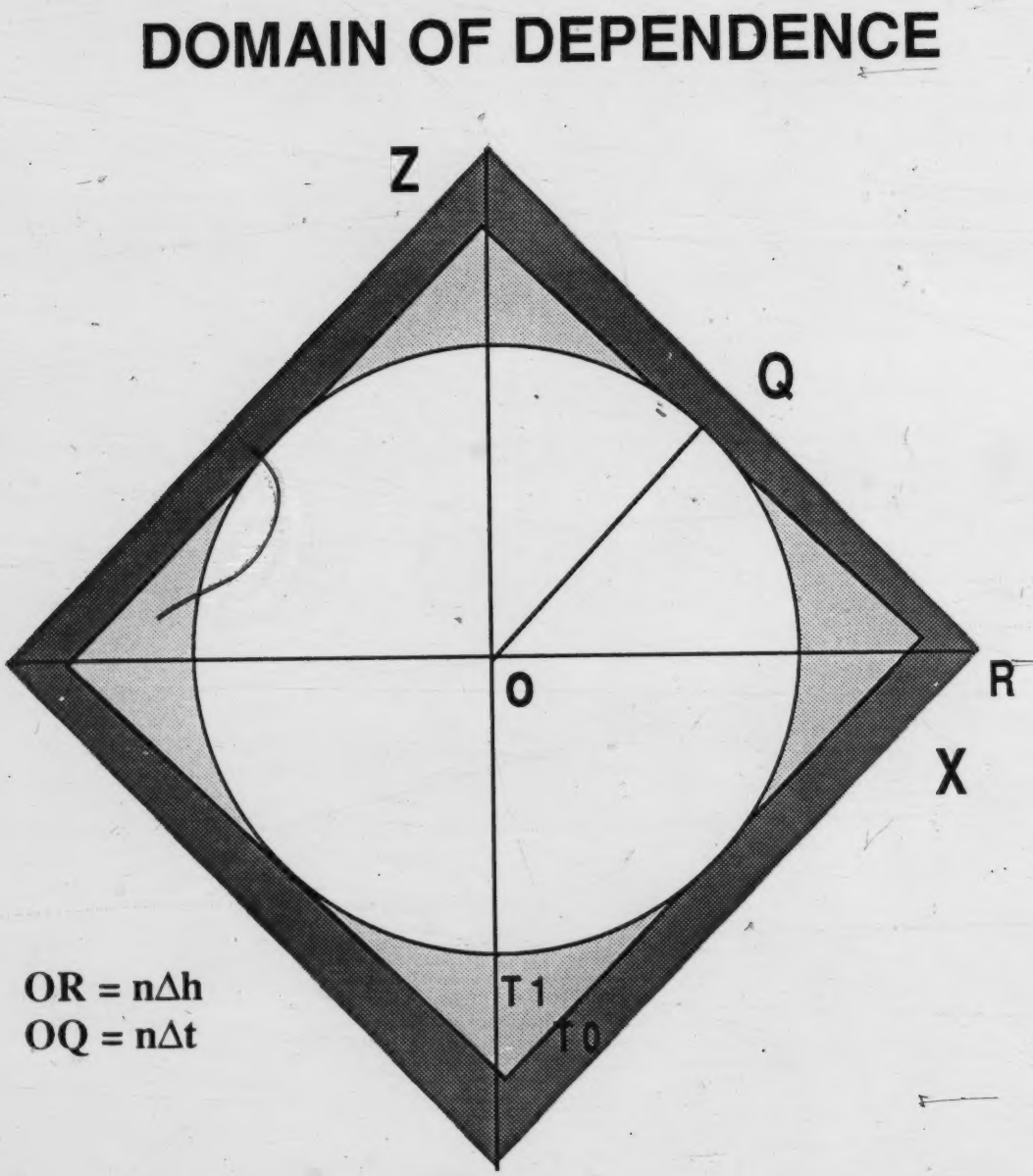


Figure 2.8. Schematic representation of the 2-D pyramidal space (domain of dependence).

2.2.6.3 Code description

When examining the migration problem which is being solved for, it was noted that a limiting factor over parallelism in time can be bypassed by the approach outlined in section 2.2.6.2. Basically, the amount of work done at each time step was too small to support large parallelism on the Myrias SPS-2.

It was realized that each point's value at time T was only related to it's previously calculated values, which could in fact suggest a partial parallelization over time. This is carried out by dividing the grid into blocks and then doing as much work as possible on an individual block within a task. For example, consider an 8 by 8 block at time T=0. After one task has been completed the values for $P_{x,z}(t)$ will be those based on the following time map:

$P_{1,5}(0) \Rightarrow$

```
00000000
01111110
01222210
01233210
01233210
01222210
01111110
00000000
```

This can be viewed as a three dimensional diagram with time running up in a pyramid toward the observer. After this task has been completed the block has values at time steps of 0,1,2,3. In general, if the block is of size 2^N by 2^N then values can be calculated by a maximum of $2^{(N-1)} - 1$ time steps. In effect, much more can be computed within a task with use of large squares. It was discovered that in order to bring all values to the same time step requires four passes (or "pardos"). Therefore $2^{(N-1)} - 1$ time steps are calculated

in only four pardos rather than the $2^{(N-1)} - 1$ "pardo's" it would take normally. This reduced the overhead quite substantially. For example with the moderately sized blocks having side lengths of 32, 15 time steps are completed with 4 pardo's.

An example with four passes will be presented. The algorithm requires that one sets up a grid, an updating function and the number of time steps to run. Then the time steps are carried in groups of $2^{(N-1)} - 1$ until finished. For each of these group of time steps, four pardo's are executed. The first pardo assumes all values to be at the same time step (for instance T=0). The system calculates upward time pyramids as follows in a 16 x 16 grid incorporating 4 pyramids:

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 1 0 0 1 1 1 1 1 1 0
0 1 2 2 2 2 2 0 0 1 2 2 2 2 2 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 2 2 2 1 0 0 1 2 2 2 2 1 0
0 1 1 1 1 1 1 0 0 1 1 1 1 1 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 1 0 0 1 1 1 1 1 1 0
0 1 2 2 2 2 2 0 0 1 2 2 2 2 2 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 2 2 2 1 0 0 1 2 2 2 2 1 0
0 1 1 1 1 1 1 0 0 1 1 1 1 1 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

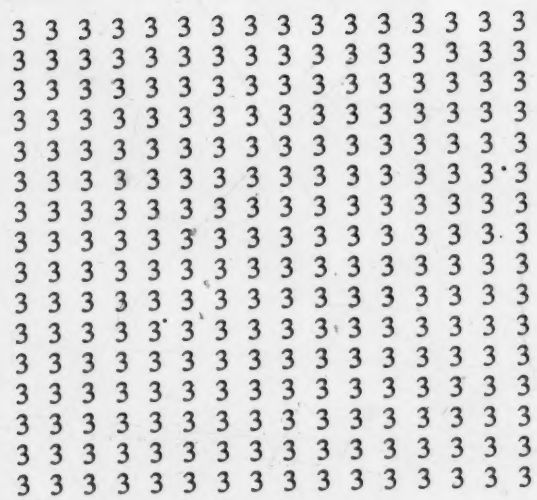
The second stage is to fill the holes amongst the vertical boxes. The grid is shifted so that the downward holes are in the center of the box. That is, one box has end points at (5,1),(5,8),(12,1),(12,8). The low areas are then filled in, creating a grid which now has columns all at the same time step.

```
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
```

The third stage is to fill the holes between the horizontal boxes. Again, this is done by an appropriate shift and filling in the low areas between boxes on the same horizontal plane.

```
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
1 1 2 3 3 2 1 1 1 1 2 3 3 2 1 1
2 2 2 3 3 2 2 2 2 2 2 3 3 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
2 2 2 3 3 2 2 2 2 2 2 3 3 2 2 2
1 1 2 3 3 2 1 1 1 1 2 3 3 2 1 1
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
1 1 2 3 3 2 1 1 1 1 2 3 3 2 1 1
2 2 2 3 3 2 2 2 2 2 2 3 3 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
2 2 2 3 3 2 2 2 2 2 2 3 3 2 2 2
1 1 2 3 3 2 1 1 1 1 2 3 3 2 1 1
0 1 2 3 3 2 1 0 0 1 2 3 3 2 1 0
```

Finally the last stage brings all values up to the same time step. This is done by filling in the holes after the appropriate shifting of the boxes. So one obtains:



This final stage can be implemented immediately before stage one thus reducing the number of pardos to three. I have programmed and tested this method on the Myrias SPS-2 for the problem of reverse - time migration. Discussions on the results of the method are shown in Chapter 3.

CHAPTER 3

2-D PRESTACK DEPTH MIGRATION IN PARALLEL

3.1 ω-X PRESTACK DEPTH MIGRATION

3.1.1 Introduction

In the first part of this chapter an examination and analysis is made of *prestack* migration of seismic data in the frequency-space domain using the 65^0 two-way scalar wave equation which has been derived and solved on a parallel computer. A near maximum degree of parallelism, efficiency and speed was achieved for the number of processors available with the new methodology used.

The seismic reflection method in the form of shot gathers provides experimental data on the elastic waves propagating through the crust and reflecting back to surface receivers. In the case of geologically complex areas, diffracted energy from outside the vertical 2-D plane (of a typical 2-D seismic survey) can create interference. In order to seek better imaging of discontinuities and resolution not available with current *poststack* algorithms, we must apply an efficient *prestack* depth migration.

Migration is the technique used to transform the reflected elastic energy from subsurface features, mispositioned by dip and interference, to their true position in the imaged section. *Prestack* migration attempts to position the seismic echoes with their true reflection amplitude at the reflecting interface and in the correct position with respect to the spatial coordinates. In order to apply this method effectively, extensive computer resources are required in terms of both power and memory storage.

The range of migration methods, making use of the scalar wave equation, can be separated into the following categories: summation or integral methods such as the Kirchhoff approach (French, 1975; Schneider, 1978); difference equations or differential methods (Claerbout 1970, 1976); and transform methods i.e *f-k* migration (Stolt, 1978; Gazdag, 1978). Berkhout (1981) gives an alternative approach using spatial convolution in the space-frequency domain. More recent work involves the implementation of the full wave or vector equation for migration schemes (Wapenaar et al, 1987).

In the presence of rapid lateral velocity variations the most reliable approach is to perform migration before stack. In this imaging technique both the shots and geophones are downward continued with images being formed at the reflector positions. A constant velocity Kirchhoff operator for this approach was applied to real data with only partial success (Sattleger and Stiller, 1974; Jain and Wren, 1980). By downward continuing common shot gathers and common geophone gathers one depth-step at a time, differential operators can be used, and vertical and lateral velocity variations can be handled (Schultz and Sherwood, 1980; Thorson, 1980). Based on the principle of reciprocity (Aki & Richards, 1980) which states that one obtains the same elastic wave form when source and receivers are interchanged, Claerbout (1985) defined this approach as *survey sinking*. A double square root expansion of the wave equation can be derived for this type of migration. The only drawback is the fact that in order to sustain reciprocity a symmetric arrangement of shots and receivers must exist, along with equally spaced shot and receiver intervals.

3.1.2 The method

The term *prestack* migration in 2-dimensions refers here to the depth migration procedure applied to shot and receiver gathers. A common *shot gather* is the seismic

recording of reflected energy from a source at a unique location to closely spaced receivers. A reverse configuration can be made of the data to form a common *receiver gather*. Migration of this form is a computationally expensive alternative to common depth point stacking (CDP) but it gives correct imaging, better dip preservation and amplitude information. The need for an accurate migration that will avoid the CDP processing artifacts is recognized by many authors leading to suggestions of methods of *prestack* migration and common source migration (Schultz and Sherwood, 1980; Reshef and Kosloff, 1986; Claerbout, 1976, 1985; Tsingas, 1990).

In seismic sounding the measurements of a physical experiment consist of common source gathers. The data of a shot-receiver gather is fully described by the wave equation for seismic waves. Obviously, this is a more complex and voluminous set of data to operate on, making the method non-feasible for serial computers. This thesis examines the use of parallel computers for prestack migration. The fact that the algorithm operates in the frequency-space domain allows all independent frequencies to be imaged in parallel.

The principle of extrapolation and imaging was suggested first by Schultz and Sherwood (1980), who solved the problem by the finite difference method in the time-space domain. Other authors (Berkhout, 1982; Stolt and Benson, 1986) have used a 45^0 approximation to obtain a more efficient algorithm and show the applicability of the method. The method applied here uses the 65^0 approximation to the scalar wave equation in the ω - x domain.

The pressure field recorded at the surface of the earth consists of shot gathers with nonzero source-receiver offsets. Therefore this type of data can not be downward continued using the "exploding reflector" concept (Lowenthal et al. 1976), which works for coincident source-receiver locations. Instead we must extrapolate the sources and receivers separately. The extrapolation is based upon the finite difference approximation of the one way acoustic wave equation carried out in two steps by a 65^0 paraxial approximation to the

scalar wave equation.

Conceptually, shots and geophones are separately downward continued into the earth, and the migrated depth section is imaged of a zero travel time and zero offset for each z-level. In general, all of the recorded data must be worked with at once. First, common shot gathers are downward continued by Δz. Then the data are reorganized into common geophone gathers and they are downward continued by Δz. In the space-frequency domain, this procedure can be carried out for all frequencies ω at the same time in parallel, and then a summation over ω will yield the imaged section. The process is then continued until z_{max} is reached.

3.1.3 Application Description

Assuming the displacement velocity is proportional to pressure and that conversion from compressional to shear waves is negligible, the observed source-receiver seismic wavefield at $z=0$ (earth's surface) can be approximated by the following scalar wave equation:

$$\frac{\partial^2}{\partial x^2} P(x_g,x_s,z,t) + \frac{\partial^2}{\partial z^2} P(x_g,x_s,z,t) = \frac{1}{v^2(x,z)} \frac{\partial^2}{\partial t^2} P(x_g,x_s,z,t) \qquad (3.1)$$

Here, x_g, x_s are the spatial geophone, source coordinates in 2-D ($x \exists x_g, x_s$), z is the depth coordinate and $v(x,z)$ is the velocity of the model medium.

The Fourier transformed field is given by:

$$P(k_g,k_s,z,\omega) = \int \int P(x_g,x_s,z,t) e^{(ik_g g + ik_s s - i\omega t)} dx\,dt \qquad (3.2)$$

Apply the operator (3.2) to the scalar wave equation to get:

$$\frac{\partial^2 P}{\partial z^2} + \left\{\left[\frac{\omega^2}{v^2(g)} - \left(k_g^2\right)\right] + \left[\frac{\omega^2}{v^2(s)} - \left(k_s^2\right)\right]\right\} P = 0 \qquad (3.3)$$

Realizing that the spatial wavenumber operator commute and are separable, the solution to this equation is the solution to the one-way scalar wave equation:

$$\frac{\partial P}{\partial z} = -i\frac{\omega}{v}\left\{\sqrt{1 - \left(\frac{v^2(g)\ k_g^2}{\omega^2}\right)} + \sqrt{1 - \left(\frac{v^2(s)\ k_s^2}{\omega^2}\right)}\right\} P \qquad (3.4)$$

Let us define as:

$$k_z(z) = \pm\frac{\omega}{v}\left[\sqrt{1 - \left(\frac{v^2(g)\ k_g^2}{\omega^2}\right)} + \sqrt{1 - \left(\frac{v^2(s)\ k_s^2}{\omega^2}\right)}\right] \qquad (3.5)$$

where k_g, k_s and k_z are the horizontal and vertical wavenumbers, and ω is the angular frequency. The choice of the minus sign in the square-root for migration is due to decrease of travel time as either geophones and shots are moved downwards.

In order to relate the above equation to shot-geophone coordinates we must start by letting the geophones descend a distance dz_g into the earth, then the equation becomes:

$$\frac{\partial P}{\partial z_g} = -i\frac{\omega}{v}\sqrt{1 - \frac{v^2(g)\ k_g^2}{\omega^2}}\ P \qquad (3.6)$$

where k_g is the horizontal wavenumber related to geophone spacing. Similarly for the downward continuation of the shots we have:

$$\frac{\partial P}{\partial z_s} = -i\frac{\omega}{v}\sqrt{1-\frac{v^2(s)\ k_s^2}{\omega^2}}\ P \qquad (3.7)$$

If we simultaneously downward continue shots and geophones by an identical amount $dz=dz_g=dz_s$ then equation (3.9) - which is the double square root equation for the downward continuation of shots and geophones - in terms of depth extrapolation can be solved by either separation or splitting because the square root operators commute ($v(s)$ commutes with k_g), so it is completely equivalent to downward continue shots and receiver separately or at the same time. Based on this conclusion equations (3.6) and (3.7) can be solved at the same depth step after they are reformulated to incorporate lateral velocity variations as described below.

3.1.4 Spatial dependance in velocity.

Let us examine the behavior of this formulation of the scalar wave equation if lateral velocity variation $v=v(x,z)$ is present. Then one may solve for $P=P(x_g,x_s,z)$ using the full acoustic wave equation. Examine the solution space in terms of equation (3.4) where the term involving the horizontal wavenumber can be replaced for example by $(ik_x^2) = \partial^2/\partial x^2$. where ($x \exists x_g,x_s$).

Bringing these equations into the spatial domain is not simply a matter of replacing a second x derivative with k_x^2 . The problem is the efficient and accurate evaluation of the square root of a differential operator. The square root is approximated by continued fractions expansion (Hildebrand, 1974 p. 501) according to:

$$R_{n+1} = 1 - \frac{f^2}{1+R_n} \qquad (3.8)$$

with $f = v(x,z)\,(k_g + k_s)/\omega$, and $R_0 = 1$, where n indicates the order of expansion. Then collecting up to second order (n=2) truncation of the continued fraction expansion, (Claerbout, 1985), equation (3.4) becomes (for the purpose of this analysis I will show only the solution for one equation as they both have the same terms) :

$$\frac{\partial P}{\partial z} = -i\left(\frac{2\omega}{v(x,z)} - \left(\frac{k_g^2}{2\frac{\omega}{v(x,z)} - \frac{v(x,z)\,k_g^2}{2\,\omega}} + \frac{k_s^2}{2\frac{\omega}{v(x,z)} - \frac{v(x,z)\,k_s^2}{2\,\omega}}\right)\right)P \qquad (3.9)$$

This is now a parabolic equation representing upgoing waves from sources and receivers. Claerbout (1985) showed that employing a retarded coordinate transformation one removes the effect of vertical translation thereby making the wave appear to be stationary. The time retarded coordinates are a system which is fixed in space relative to the ordinary Cartesian coordinate system. The retarded coordinate system is related through the local velocity, $v(z)$ at each source location to the Cartesian system by the following set of equations:

$$\begin{aligned} t' &= t'(x,z,t) = t - 2\int_0^z dz/v(z) \\ x' &= x'(x,z,t) = x \\ z' &= z'(x,z,t) = z \end{aligned} \qquad (3.10)$$

The pressure wavefield can be written in terms of a stationary wave $Q(x_g,x_s,z,\omega)$ in the frequency domain as:

$$P(x_g,x_s,z,\omega) = Q(x_g,x_s,z,\omega)\; e^{-2i\,\omega\int_0^z \frac{dz}{v(z)}} \qquad (3.11a)$$

or

$$Q(x_g,x_s,z,\omega) = P(x_g,x_s,z,\omega)\; e^{2\,i\,\omega\int_0^z \frac{dz}{v(z)}} \tag{3.11b}$$

Differentiating (3.11a) with respect to z gives:

$$\frac{\partial\, P(x_g,x_s,z,\omega)}{\partial z} = (\frac{\partial}{\partial z} - \frac{i\omega}{v(z)})\; Q(x_g,x_s,z,\omega)\, e^{2\,i\,\omega\int_0^z \frac{dz}{v(z)}} \tag{3.12}$$

Substitute (3.11a) and (3.12) into (3.9) so that we obtain the time shifted downgoing wavefield:

$$\frac{\partial Q(x_g,x_s,z,\omega)}{\partial z} = +\,i\left(\frac{k_g^2}{2\frac{\omega}{v(x,z)} - \frac{v(x,z)\;k_g^2}{2\;\omega}} + \frac{k_s^2}{2\frac{\omega}{v(x,z)} - \frac{v(x,z)\;k_s^2}{2\;\omega}}\right) Q(x_g,x_s,z,\omega)$$
$$-\,2i\;\omega(\frac{1}{v(x,z)} - \frac{1}{v(z)})\; Q(x_g,x_s,z,\omega) \tag{3.16}$$

The first term on the right hand side represents the *diffraction term*. The second term on the right is called the *thin lens* term. In the derivation above we assume that v(x,z) is a locally constant medium velocity and v(z) depends on depth below the source, only.

Upon splitting the *diffraction term* becomes:

$$\frac{\partial Q(x_g,x_s,z,\omega)}{\partial z} = +\,i\left(\frac{k_g^2}{2\frac{\omega}{v(x,z)} - \frac{v(x,z)\;k_g^2}{2\;\omega}}\right) Q(x_g,x_s,z,\omega) \tag{3.17a}$$

and

$$\frac{\partial Q(x_g,x_s,z,\omega)}{\partial z} = +i\left(\frac{k_s^2}{2\dfrac{\omega}{v(x,z)} - \dfrac{v(x,z)\ k_s^2}{2\ \omega}}\right) Q(x_g,x_s,z,\omega) \qquad (3.17b)$$

Equations (3.17a) and (3.17b) may be rearranged by substituting for:

$$-k_g{}^2 = \partial^2/\partial g^2,\ -k_s{}^2 = \partial^2/\partial s^2 \text{ and } \mu(x,z)=\omega/v(x,z)$$

Then the diffraction portions of the solution are given by the following approximation:

$$-\frac{i\,\mu(x,z)}{\alpha}\frac{\partial Q(x_g,x_s,z,\omega)}{\partial z} - \frac{i\,\beta}{\alpha\mu(x,z)}\frac{\partial^3}{\partial g^2\,\partial z} Q(x_g,x_s,z,\omega) + \frac{\partial^2}{\partial g^2} Q(x_g,x_s,z,\omega) = 0 \quad (3.18a)$$

and

$$-\frac{i\,\mu(x,z)}{\alpha}\frac{\partial Q(x_g,x_s,z,\omega)}{\partial z} - \frac{i\,\beta}{\alpha\mu(x,z)}\frac{\partial^3}{\partial s^2\,\partial z} Q(x_g,x_s,z,\omega) + \frac{\partial^2}{\partial s^2} Q(x_g,x_s,z,\omega) = 0 \quad (3.18b)$$

where for the 65 - degree approximation of the square root the coefficients are (Yilmaz,1987):

α=0.478242060 and β=0.376369527.

These equations will be evaluated step by step starting at the surface and moving to any depth using a finite difference approximation. The thin lens term may be solved analytically and is incorporated as a correction within each depth step prior to the evaluation of equation (3.18) and is given by:

$$\frac{\partial Q(x_g,x_s,z,\omega)}{\partial z} = i\,\omega\left(\frac{1}{v(x,z)} - \frac{1}{v(z)}\right) Q(x_g,x_s,z,\omega) \qquad (3.19)$$

Equations (3.18) is also called the 65^0 approximation to the scalar wave equation, since this form of the square root approximation to the dispersion relation yields reliable results when the angles of incidence of the rays normal to the wavefront are less than about 65^0 or 70^0. As mentioned previously, it is also called a paraxial equation and has all the properties of parabolic differential equations. It is the main equation for ω-x migration.

Converting equations (3.18a and 3.18b) to a finite difference form (Appendix B) and writing this in matrix form for each harmonic yields an independent set of tridiagonal solutions for each harmonic component, ω_n.

Expressing the wave equation in this form allows one to see the flexibility that exists in order to propagate the wavefield in terms of each frequency in parallel, solving for all necessary depth steps, and summing at the end. Equation (3.18) can also be included in the same task (processor) because it simply shifts Q at each frequency simultaneously. Some parallel architectures might offer the potential of automatic merging (summing of an array) which would reduce the computational time in these cases.

The code can be extended to include density variations and higher order accuracies of the square root approximation. Also an extension of the code to include the effects of transversely isotropic media has been analyzed (Phadke et al. 1991). The application of this ω-x algorithm for migration is described below.

We have seen that decomposition of the wave equation in terms of monochromatic plane wave propagation at different angles allows one to extrapolate in depth each temporal frequency harmonic component which is part of the wave solution or task. No intertask communication (overlapping) is necessary during the extrapolation as each harmonic propagates as an independent mode. This method shows a great degree of parallelism.

The efficiency of its implementation on a MIMD computer will be seen below.

The final step in the method is to apply the imaging condition by summing the $Q(x_g,x_s,z)$ field over all frequencies:

$$Q(x_g,x_s,z,t=0) = \frac{1}{N} \sum_{\omega} Q(x_g,x_s,z,\omega) \qquad (3.20)$$

The necessary steps for shot gather migration and the logistics of its parallel implementation are described below.

3.1.5 Parallel algorithm implementation

The restructuring of this migration code in parallel was carried out as follows. Based on the linear superposition of monochromatic harmonics in wave propagation one may compute each frequency component of $Q(x_g,x_s,z,\omega_i)$ separately during steps (3.11) and (3.12). Keeping these in mind the algorithm was designed as follows:

1. Temporally transform the wavefield Q by calculation the Fourier transform of all shot records in parallel.

2. For each frequency step in parallel

2.1 for all depth steps

A. for all shot records (in parallel if more processors are available)

a. Estimate the time retarded field Q using (3.11a).

b. For each depth step solve (3.18a) and (3.18b).

c. reorder the data in receiver gathers

B. for all receiver gathers (in parallel if more processors are available)

d. repeat steps (a) and (b) until all depth points are resolved.

3. Automatic summing takes place over all depth steps in parallel for all the frequency components to get the final migrated section.

(Equation (3.19) is applied only if there is lateral heterogeneity present)

A description of the pseudocode is shown in Figure 3.1, and a graphical representation of the process in Figure 3.2.

3.1.6 Performance monitoring on the Myrias SPS-2

The 2-D *prestack* ω -x algorithm was tested extensively on the SPS-2 using from 64 PE's (processing elements) to 1025 PE's. In the initial tests a small data set of 96Kb was examined. The run-time was less than 50 seconds on 64 PE's, achieving four times the performance of an Amdhal 5870.

A larger data set was tested using 30 Mb of data running 60 shot records over 600 depth steps per shot. Using 64 frequency components on 64 PE's the algorithm applied in frequency tasks takes about 7.5 hours. The same run using 128 PE's achieved more then 90% efficiency, and about 3.5 hours elapsed time. The problem was also tested on 1025 PE's, using 256 frequency component tasks per shot record. The total elapsed time to migrate 60 shot records over 600 depth steps, was 3303 seconds (~ 50 min). A similar problem running on an IBM 3090 (without vector facility) took approximately 60 hours (D. Ganley, Canterra, personal communication).

The time performance of the method is shown in Table 3.1, with observed times using a range of PE's shown in Figure 3.3.

TABLE 3.1 Performance monitoring ω-x migration

PE's	*Shots*	*Frequencies*	*Time(sec.)*
64	*1*	*256*	*787*
64	*60*	*64*	*12600*
256	*10*	*128*	*1030*
296	*20*	*256*	*3628*
1025	*60*	*256*	*3303*

From the results above, the migration code has shown a dynamic degree of scalability over the number of PE's. Since each shot record and each frequency component can be estimated independently, the automatic load balancing of the SPS-2 resulted in an efficiency of more than 90% over the execution time. A speed-up graph over the number of PE's is shown in Figure 3.4, which depicts the linear increase in speed over the basic 64 processors both in theory and observation. There is more than enough work to saturate a very large parallel computer.

Based on the performance of the algorithm on the SPS-2, tested on a relatively small seismic data set, excluding delays of porting seismic data onto the system (reading tapes and converting IBM to IEEE formats), the time span to get migrated results was satisfactory. Being able to achieve a day-to-day turn-around time for *prestack* migration, the SPS-2 has shown a potential for more accurate seismic applications. Another point to consider is that the programmability of the SPS-2 made it easy to capture the parallelism inherent in seismic processing.

S-G ω-x MIGRATION PSEUDOCODE

(MAIN PART OF PARALLELISM)

(PARALLEL) DO over Frequency steps

DO over Depth steps

(PARALLEL) DO over number of shots

A. Calculate the time retarded field

$$Q(x,z,\omega) = P(x,z,\omega)\ e^{2i\,\omega \int_0^z \frac{dz}{v(z)}}$$

B. Downward continue the wave field using the diffraction term

$$-\frac{i\,\mu(x,z)}{\alpha}\frac{\partial Q(x,z,\omega)}{\partial z} - \frac{i\beta}{\alpha\mu(x,z)}\frac{\partial^3}{\partial x^2\,\partial z}Q(x,z,\omega) + \frac{\partial^2}{\partial x^2}Q(x,z,\omega) = 0$$

C. and the thin lens term

$$\frac{\partial Q(x,z,\omega)}{\partial z} = -\,i\,\omega\left(\frac{1}{v(x,z)} - \frac{1}{v(z)}\right)Q(x,z,\omega)$$

END DO over shots

(PARALLEL) DO over number of receivers
REPEAT (A -B -C)
END DO over receivers

END DO LOOP over depth

END PARDO over frequencies

Figure 3.1. Pseudocode of the Omega-x migration as implemented in parallel.

ω – x MIGRATION IN PARALLEL

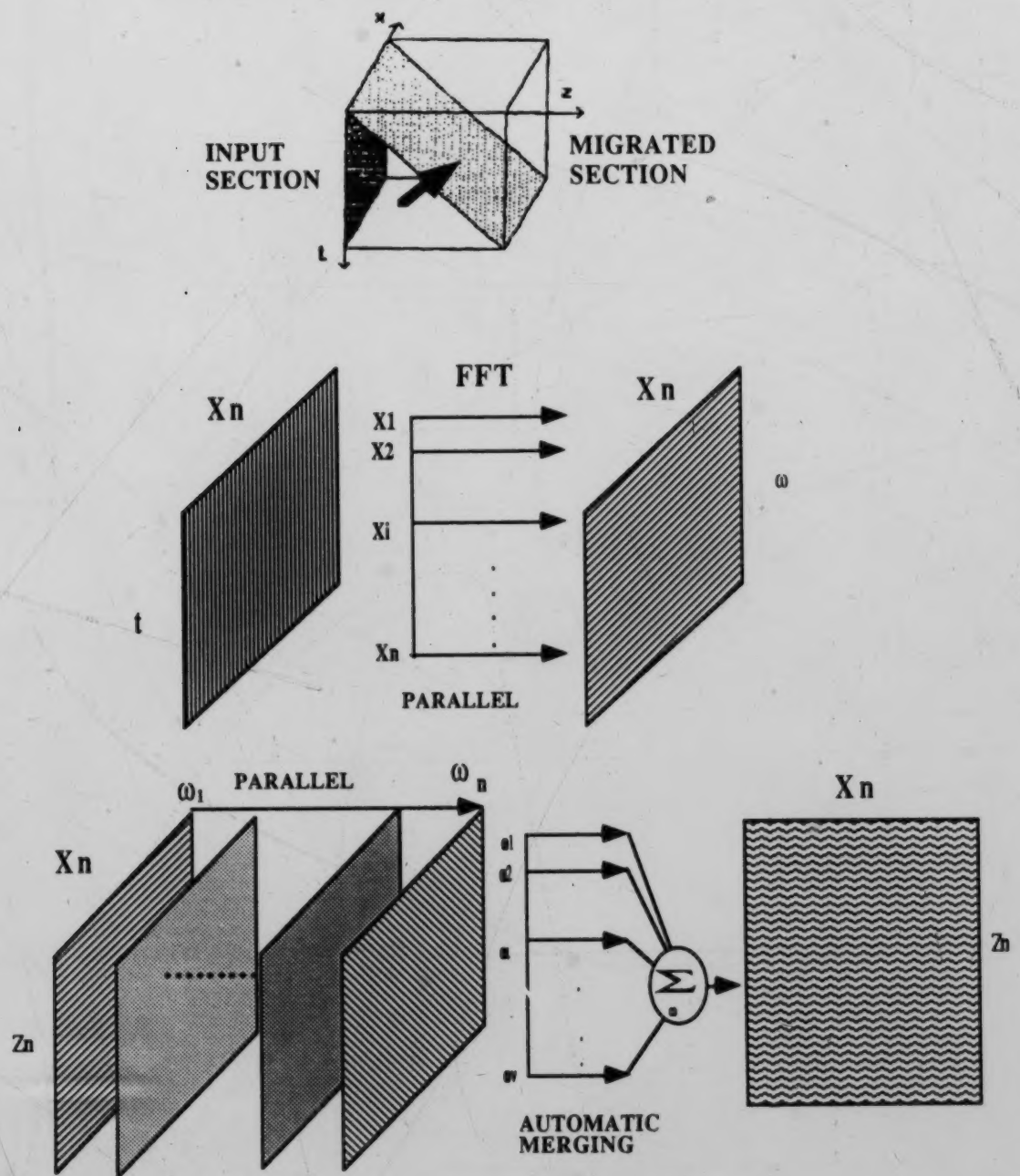


Figure 3.2. Graphical representation of the Omega-x migration.

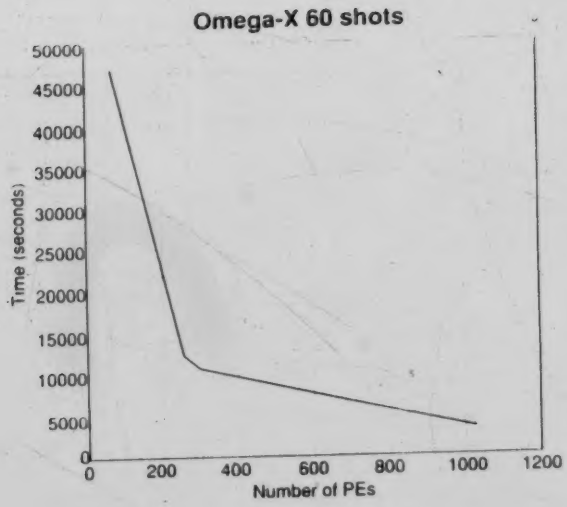


Figure 3.3. Time vs PE's for omega - x migration.

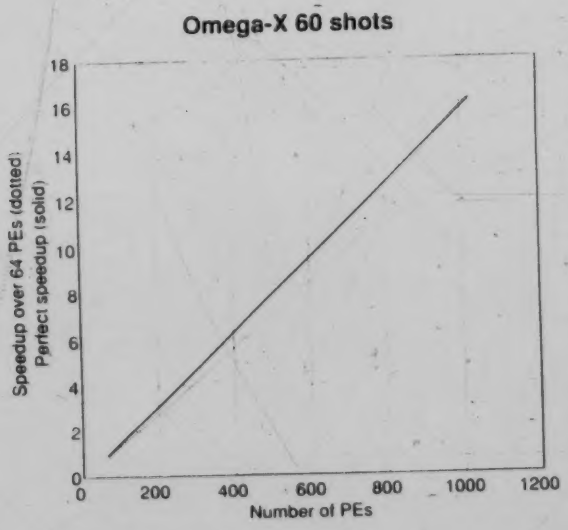


Figure 3.4. Speed up over 64 PE's. Solid line theoretical, dotted line experimental.

3.1.7 Analysis and Applications

3.1.7.1 Difference operator response

In order to examine the accuracy of the difference operator based on the degree of approximation (i.e. 65^0) we need to test the depth migration response due to a time impulse as shown in Figure 3.5a. For a constant velocity medium the difference operator response should form a semicircle in the x-z space with a radius of z=v*t, where v is the velocity of the medium and t is the travel time. In terms of 90^0 degree accuracy approximation the response should be a perfect semicircle (Figure 3.5a).

The time impulse in this test is at 0.3sec. with a homogeneous velocity of 1000 m/s for a model section with lateral spacing of 40 m. The responses of 15^0, 45^0 and 65^0 approximations are shown in Figure 3.5b,c,d. It can be seen that the 65^0 solution approximates the semicircle with higher accuracy than the other two. This level of approximation was used during this analysis and was considered accurate for the degree of dip present in our data.

3.1.7.2 Synthetic examples

The ω-x migration algorithm was tested extensively with various sets of synthetic data. Originally a synthetic example of seismic responses generated over a subsurface step fault (Figure 3.6) model was tested. Part of the input shot data is shown in Figure 3.7, generated using a ray-tracing algorithm (Phadke et al. 1990) with receiver spacing of 50m. with 25 receivers per shot and a sampling interval of 2 ms. The result of the *prestack* migration is shown in Figure 3.8. Figure 3.9 shows the migrated stack of these shot records. It can be seen that the method images the response correctly revealing the character of the model. The resolution of the result is greater at the near offset, simply because the method is more accurate at those offsets.

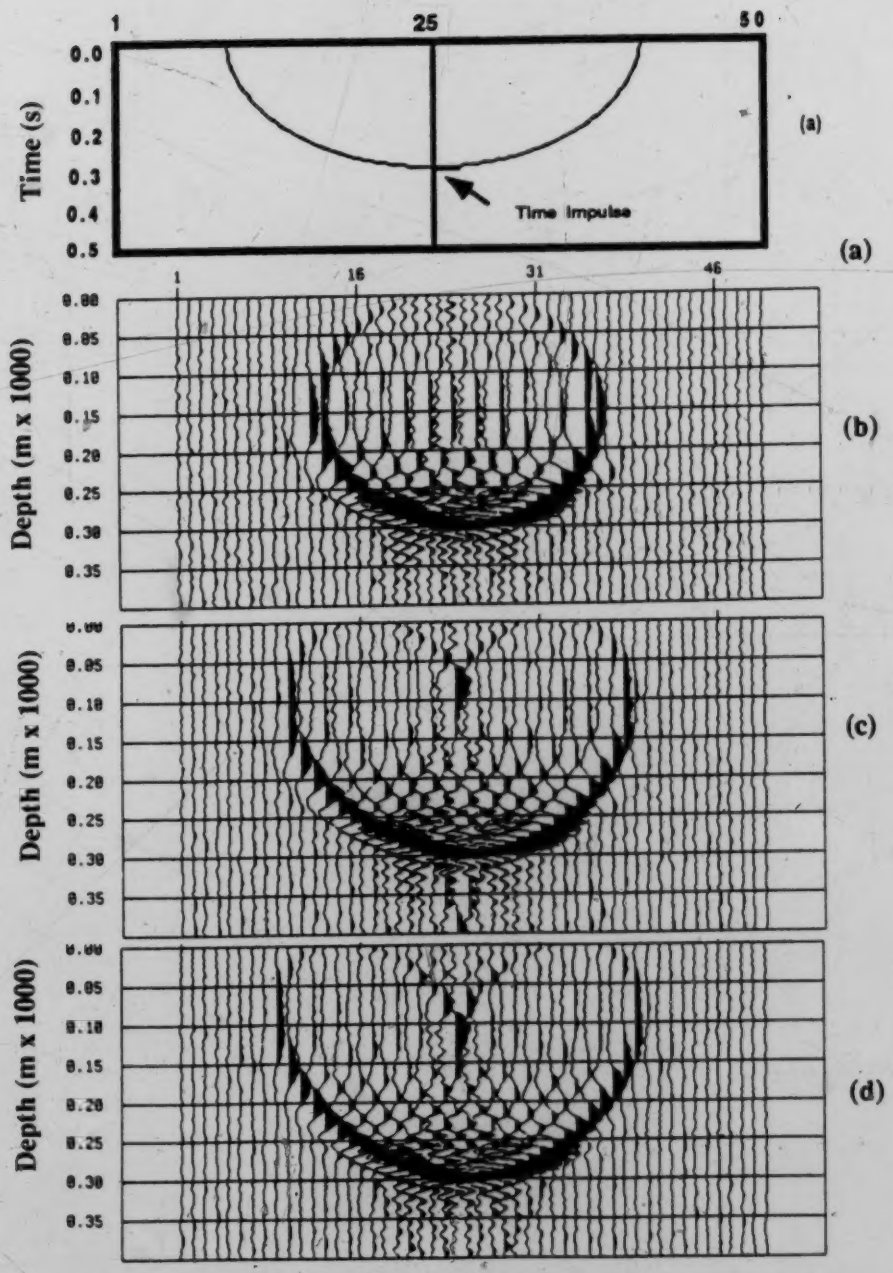


Figure 3.5 (a) Desired impulse response for migration, (b) 15^0 approximation, (c) 45^0 approximation and (d) is the 65^0

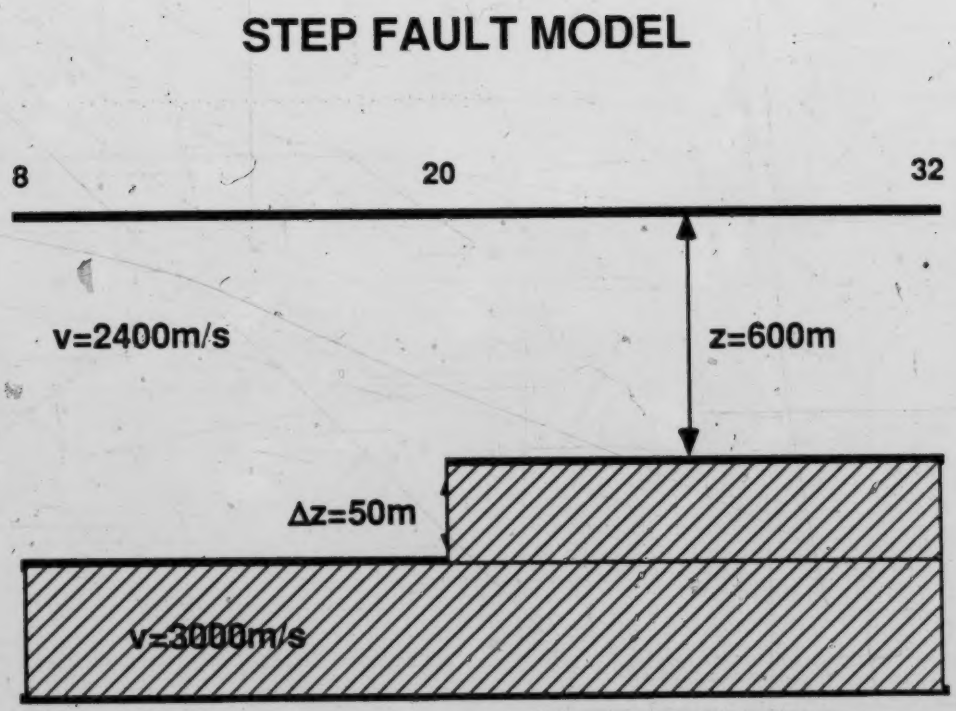


Figure 3.6 Step Fault Model used to test the prestack migration method.

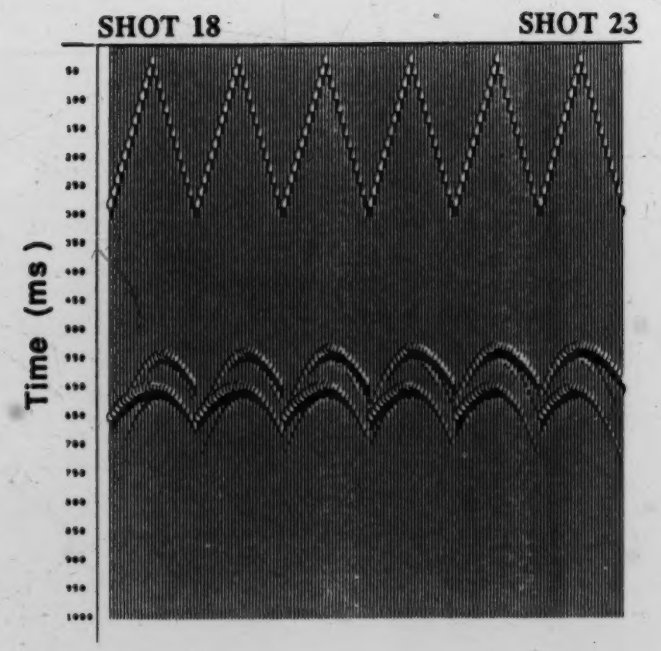


Figure 3.7 Generated synthetic shot gather for step fault model via ray tracing.

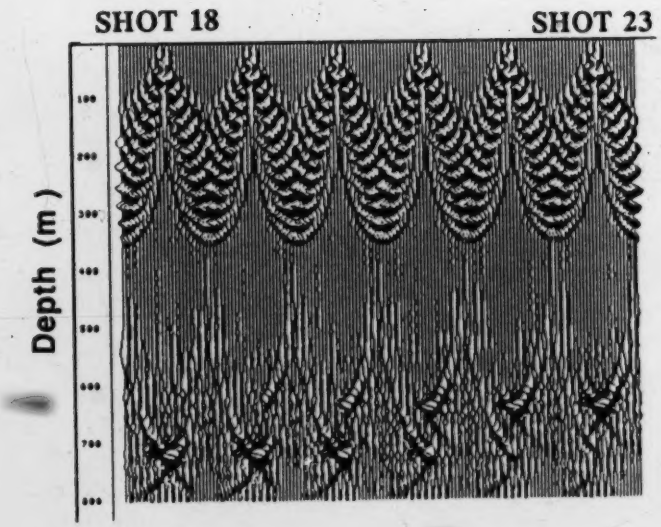


Figure 3.8 Prestack depth migrated shot record for step fault model.

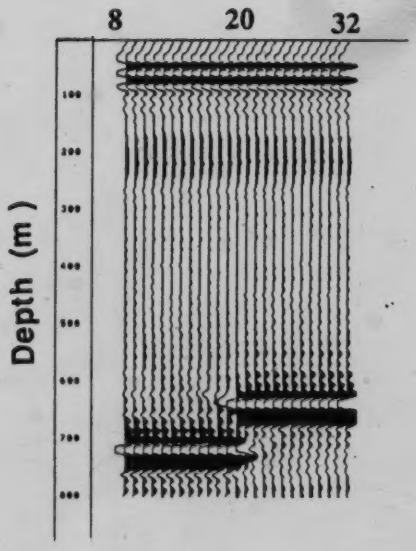


Figure 3.9 Final prestack depth migrated stack of the fault model. The character of the fault is clear justifying the accuracy of the method.

A second case was examined using synthetic data. The model is a complex one involving many faults and changes in velocity (Figure 3.10). The input data, generated by an Aimes commercial algorithm using raytracing and provided by Chevron Resources Ltd., was comprized of 60 shot records over the modeled region with 120 records each at a spacing of 24m. at a sampling interval of 1ms. The set was prestack depth migrated with a depth resolution of 4 meters. Figure 3.11 shows the synthetic shots records over this model and Figure 3.12 their migrated counterparts. The final migrated stack is shown in Figure 3.14. and the Common Depth Stack (CDP) in Figure 3.13. The color plots represent the seismic wave amplitude variations in the section. It can be seen that the method images correctly the character of the faults as well as the graben on the right side of the figure. The CDP section only reveals the diffractions due to faulting and does not image the true reflector positions.

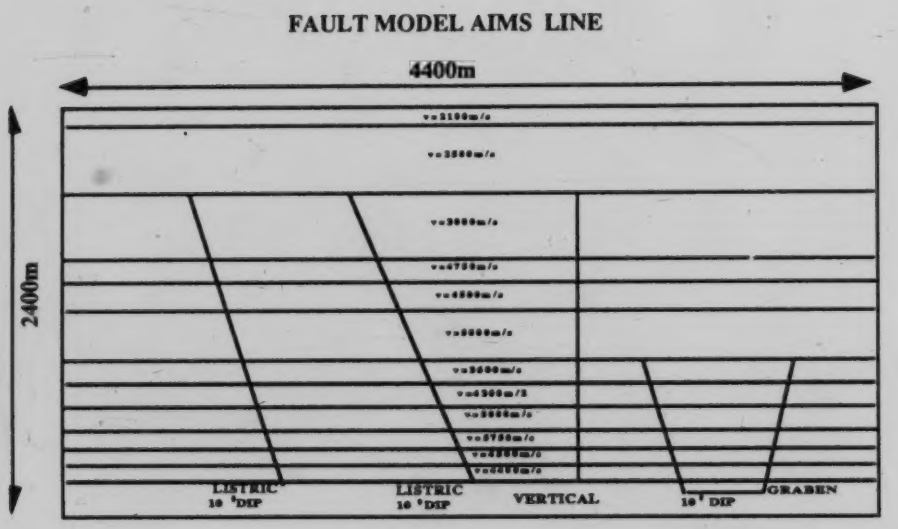


Figure 3.10 Geological model used to generate synthetic seismic records.

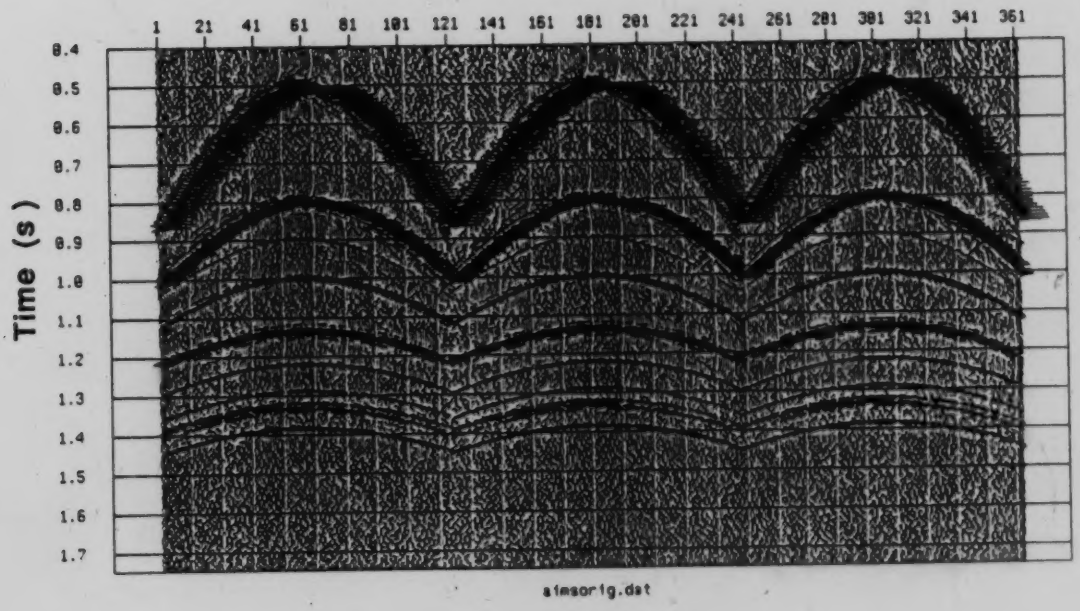


Figure 3.11 A few representative examples of synthetic shot records for the model in Figure 3.10.

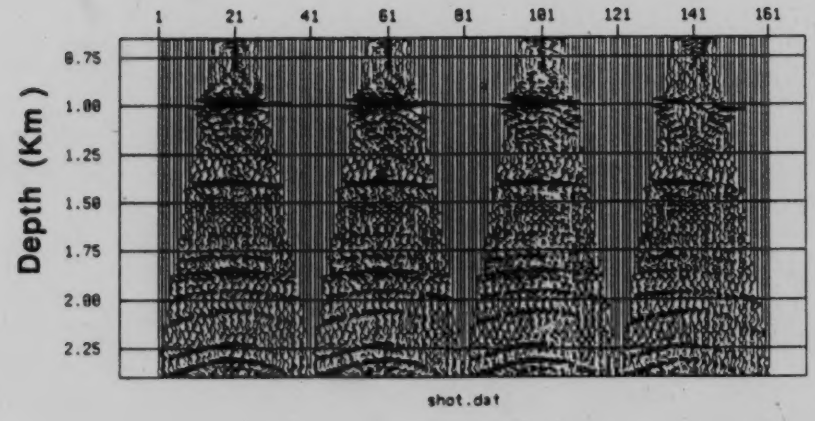


Figure 3.12 Prestack depth migrated shot gathers from Figure 3.11.

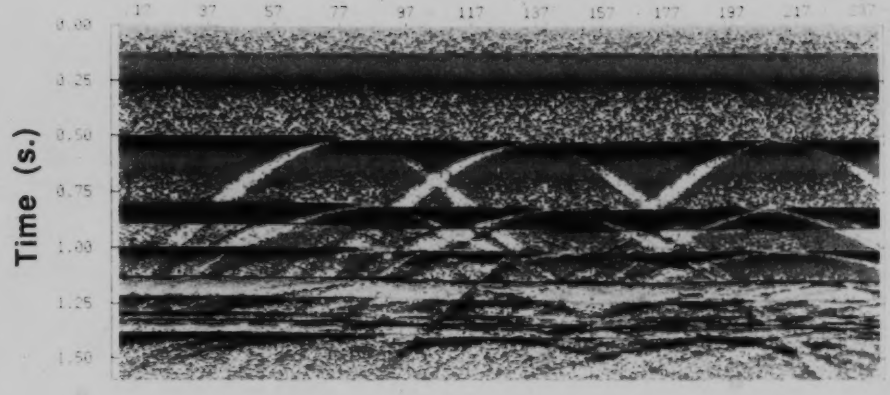


Figure 3.13 Common Depth Point stack for the model in Figure 3.10.

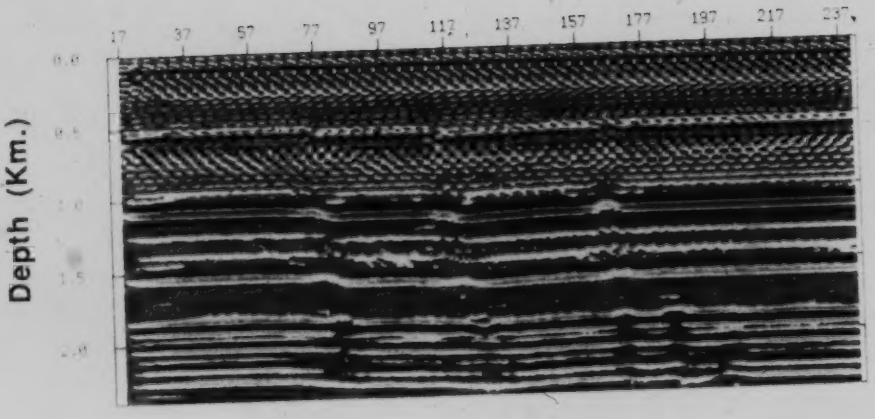


Figure 3.14 Prestack depth migrated stack of the model in Figure 3.10

3.1.7.3 Real data example

In order to test the validity of the algorithm a seismic reflection data set from an east-west line south of Princess well in Alberta was chosen for real data. Figure 3.15 shows the location map of the line. The data was acquired by Pan-Canadian Petroleum Ltd. in March 1985, and a small portion of this line (approximately 6800m. in length) was made available for prestack depth migration. The seismic instrumentation consisted of a 120-channel DFS-V recording system and air-gun sources. An inline split-spread shooting pattern with a shotpoint interval of 34 m and a receiver group interval of 17 m was used. Each receiver group comprised 9 geophone units in an inline pattern.

A CMP stack of these data with a maximum of 30-fold coverage is shown after conventional processing in Figure 3.16. A very strong reflector (the top of the erosional surface of Missisipian deposits) at a two-way reflection time of 0.83s. can be seen on this section. The reflectors between 1.0 and 1.4 s. are from Cambrian formations. The Precambrian crystalline basement is probably at 1.4 s. and is relatively continuous throughout the section except for the interruption by diffraction arcs. The diffraction energy is remarkable on this section and has not been completely removed by CMP stacking. If the diffractions were not present, it would be difficult to infer any discontinuities in the reflecting horizons. In order to locate discontinuities due to small faults or sudden changes in lithology in the area, prestack depth migration was applied to these data.

Before the application of prestack migration a global trace balancing was applied to the data using the following method: a time window for 0.8 to 1.0 s. was selected from one shot record and the mean of the amplitudes squared was computed for all 120 traces in the gather, resulting in a global factor. The reason for the selection of this time window is that within this interval the reflections are consistent and continuous. Then, for each trace in

each shot gather, the sum of amplitudes squared within the same window (0.8 to 1.0s.) was computed, resulting in a trace factor. The ratio between the global factor and the trace factor was applied to each trace of all shots in the line. Consequently, all the source records have been normalized to a uniform amplitude. Also, in order to satisfy the principle of reciprocity of the migration method, every second trace on the record was chosen to result in 60 traces per shot so that an equidistant shot and receiver spacing of 34 m. can be sustained.

A shot record and a prestack migrated one are shown in Figure 3.17. The final prestack migrated stack is shown in Figure 3.18. An important feature which can be seen on the left side of the Figure and at a depth between 2 to 2.4 km. can be attributed to an explanation given by Kanasewich (1968) and also verified by the CFP (common fault point) method used by Kanasewich and Phadke (1988).

This data set is from a northern edge of major east-west rift of Precambrian origin (Kanasewich, 1968). It is also 6.4 km south of a well drilled through 1874 m of Phanerozoic sediments into a basement high with the basal core being a mafic metamorphic rock (Burwash, 1987). The anomaly has a positive gravity expression and is presumed to be volcanic extrusive through middle Cambrian sediments as part of the rifting process. The anomalous diffraction patterns on the CDP seismic section (Figure 3.16) interrupting Middle Cambrian Arctomys, Pika, and Stephen formations are interpreted as representing igneous intrusives (a in Figure 3.18) with some possible reactivation (b) and faulting (d) during subsequent Paleozoic time, as has been also interpreted by (Kanasewich and Phadke, 1988). Other faulting patterns may be due to erosional features or carbonate buildups. In conclusion, the prestack depth migration method was able to delineate these features and further justify the original interpretation of the geological phenomena in the region.

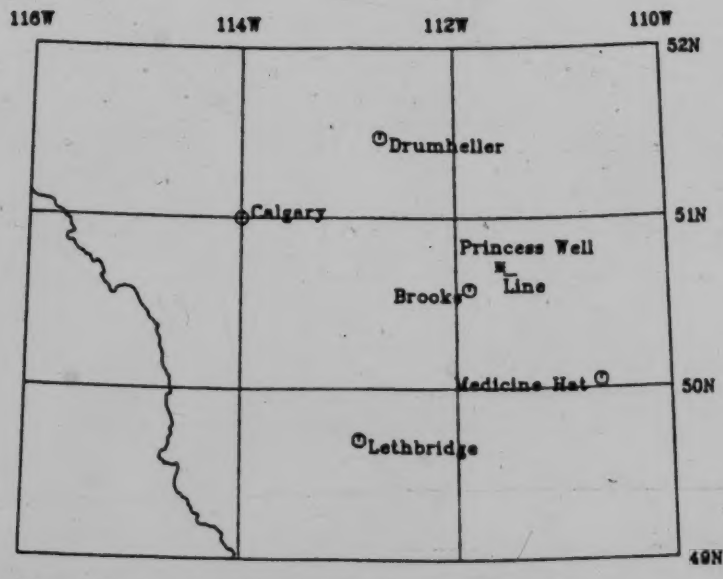


Figure 3.15 Map of the location showing the seismic line in Southern Alberta.

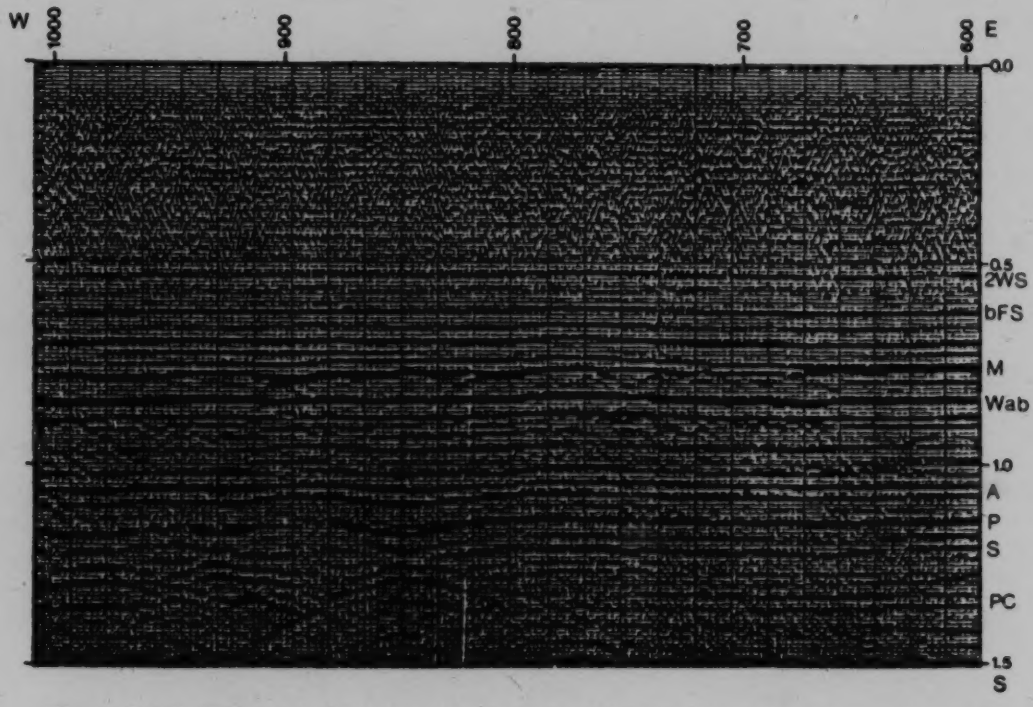


Figure 3.16 CDP stacked section of a portion of the line from the Princess well area.

(From Kanasewich and Phadke, 1988)

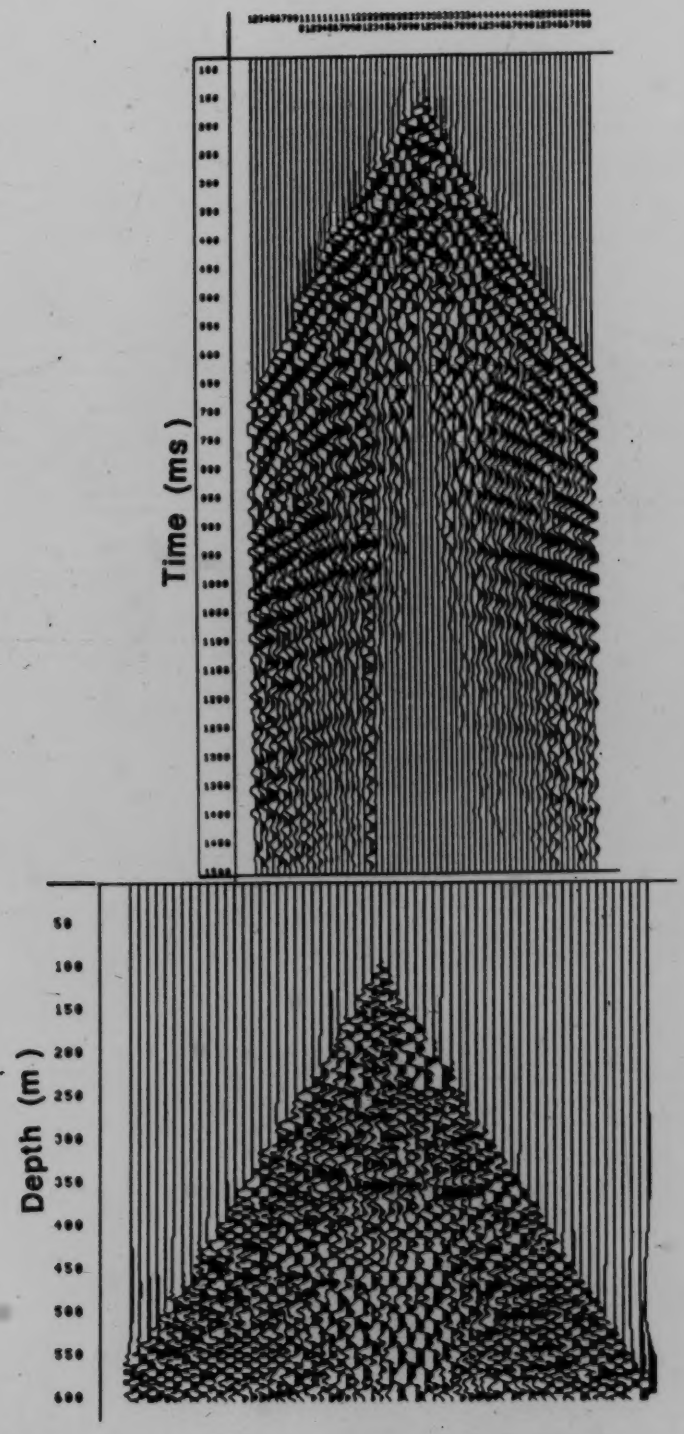


Figure 3.17 Shot record (a) and prestack migrated counterpart (b).

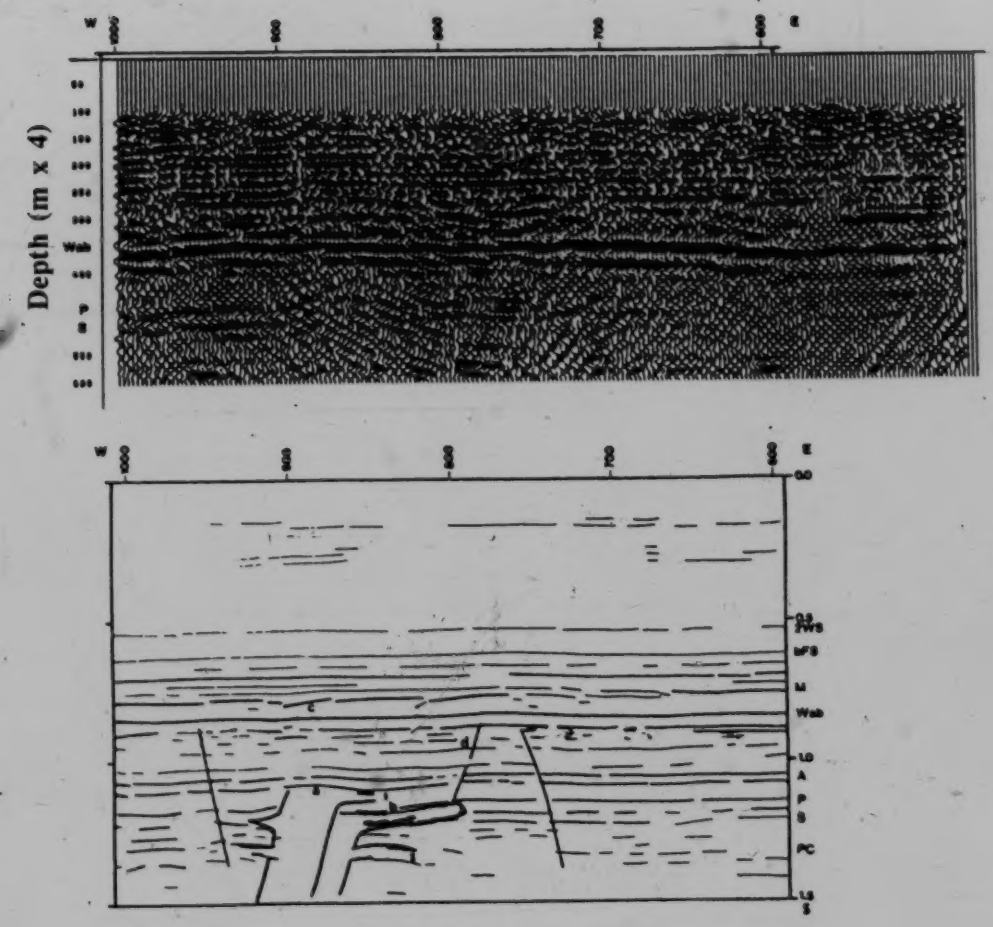


Figure 3.18 Prestack depth migrated stack of the Princess Well Data and Interpreted section from Kanasewich and Phadke (1988)

3.2. REVERSE TIME FINITE-DIFFERENCE PRESTACK MIGRATION

3.2.1 Introduction

A *prestack* migration algorithm for acoustic waves in two-dimensional variable velocity media has been developed, implemented and tested on a parallel computer. The algorithm operates in the time-space domain and is based on reverse-time finite-difference extrapolation of acoustic waves. The reverse time migration process is well known in the literature (Sun and McMechan, 1986; Baysal et al, 1983;, McMechan, 1983). Most of the authors apply ray-tracing or forward source extrapolation to achieve an imaging condition. The method introduced here applies reverse time boundary information on the difference grid and uses the same principle of double-downward continuation as in the case of the ω-x domain. In this analysis the problem is solved in both space parallelism and partial time parallelism.

3.2.2 Application description

Solutions to the wave equation by finite-difference methods have been used widely in the literature (Boore, 1970; Alford and Kelly, 1974). These methods are good for structurally complex subsurface geometries. Geometries of particular interest to petroleum exploration are those containing sharp corners which generate diffractions. The purpose of this work is to implement finite difference schemes in a parallel computer and test with typical geophysical problems.

The governing equation describing the pressure field $P(x_g,x_s,z,t)$ in a homogeneous region due to a source distribution located at the surface is:

$$\left[\nabla^2 - \frac{1}{C^2}\frac{\partial^2}{\partial t^2}\right] P(x_g, x_s, z, t) = 0 \qquad (3.21)$$

in the homogeneous form, where C(x,z) is the velocity of acoustic propagation in the medium, and ∇^2 is the Laplacian operator.

If we split equation (3.21) into shot and receiver gather domains then it can be approximated by an explicit second-order difference scheme (Mitchell, 1969) also described earlier in Chapter 2:

$$\begin{aligned} P(m,n,l+1) &= 2(1-2q^2)P(m,n,l) \\ &+ q^2[P(m+1,n,l) + P(m-1,n,l) + P(m,n+1,l) + P(m,n-1,l)] \\ &- P(m,n,l-1) + O(h^2 + \Delta t^2) \end{aligned} \qquad (3.22)$$

where $\Delta x = \Delta z = h$ is the grid size in the x (x_g or x_s) and z directions, respectively; Δt is the time step; m,n,l are integers such that $x=m\Delta x$, $z=n\Delta z$, $t=l\Delta t$; $q=C\Delta t/h$ and $O(h^2)$ indicate the scheme approximates the corresponding partial differential equation to order h^2 . Another expression can be used to obtain a more accurate fourth order representation which looks as follows (Alford and Kelly, 1974):

$$\begin{aligned} P(m,n,l+1) &= (2-5q^2)P(m,n,l) \\ &+ \frac{q^2}{12}\{16[P(m+1,n,l) + P(m-1,n,l) + P(m,n+1,l) + P(m,n-1,l)] \\ &- [P(m+2,n,l) + P(m-2,n,l) + P(m,n+2,l) + P(m,n-2,l)]\} \\ &- P(m,n,l-1) + O(h^4 + \Delta t^4) \end{aligned} \qquad (3.23)$$

Further to the above formulations one should be aware of the limitations of stability of these schemes, Equation (3.6) is known to be stable (Mitchell, 1969) if:

$$q \le \frac{1}{\sqrt{2}} \qquad (3.24)$$

Similarly for the fourth order scheme stability is bounded by:

$$q \leq \sqrt{\frac{3}{8}} \tag{3.24}$$

In the following section an examination of the principle of double downward reverse time migration based on the wave extrapolation using the above difference equations will be discussed.

3.2.3 Extrapolation and imaging of acoustic waves

The prestack reverse time migration method developed here consists of two parts; reverse extrapolation of the recorded waves from the recorder position back into the medium, and downward extrapolation of the wave from the source into the medium. When the two waves meet, imaging has been achieved. This section will give some details of these applications and then relate them to the design of the algorithm.

Reverse time extrapolation is a boundary value problem in which the acoustic finite difference meshes *(x,z)* are driven respectively with the time reverse of the field observed at each recorder. At a specific time step t_i during extrapolation of a set of data; the energy in the seismograms at times $> t_i$ has already been transferred to the finite difference *x-z* meshes and is propagating back toward the point at which the disturbance was generated.

At time t_i the pressure values along the time slice are inserted as a boundary at the corresponding recorder positions in the difference grid. Then, once one finite-difference step is taken, new boundary values are inserted from the data at t_{i-1}, and so forth. In this manner the wave moves with its assigned velocity and coalesces to a single point (in the case of a diffractor).

The determination and application of the imaging condition at each point in time

requires double continuation of sources and receivers downwards at the same time-step. Since imaged energy is coincident with the direct wavefront from the primary source this method will satisfy the imaging condition everywhere.

A brief description of the application is given here. As time propagates backwards through the grid, we reorder the data in receiver gathers and downward continue the sources. Using this concept amplitudes are correlated (constructive interference) on the difference grid along the image.

The proper design of a parallel algorithm to operate in this space was shown in Chapter 2. The analysis has been broken up into two cases. Case one described the initial formulation based on a coarse parallelism scheme working in serial over time. The second case involved the advanced parallelism design of the code to operate in partial parallel time steps

The basic limitations of the algorithm are due to the fact that Δx should be equal to Δz which restricts the applicability of the method unless an appropriate seismic survey exists. We limit our analysis here to synthetic seismic records that obey this rule, but one can improve the difference scheme on the code to account for more general cases.

3.2.4 Performance monitoring on SPS

The reverse time migration codes were both tested on sets of seismic data in order to check their validity, speed and efficiency. The coarse parallelism code was run in both pre- and post- stack domains using second and fourth order difference schemes. The advance parallel algorithm is applied in a poststack domain using second order differences.

The time performance of the coarse parallelism method is shown below. It was tested on prestack data consisting of 48 traces, 150 depth steps and 500 time samples, and also on poststack data consisting of 512 traces, 300 depth steps and 1500 time samples using various number of processors.

TABLE 3.2 Performance monitoring coarse F-D migration

Problem size	*Processing Elements*						
	1	*2*	*8*	*16*	*25*	*50*	*100*
150 x 48	*430*	*360*	*289*	*221*	*160*	*107*	*(times in seconds)*
300 x 500						*1400*	*890*
						(45% efficiency)	
200x39x25	—			*3609*		*1326 (60% efficiency)*	

The major overheads involved in this problem are due to the inherently sequential requirements of stepping over time. Time performance of the advanced method applied to a poststack record of 256 traces, 256 depth steps and 1500 time steps is shown in the following table:

TABLE 3.3 Performance monitoring advanced F-D migration

Problem size	*Processing Elements*		
	32	*64*	*128*
256 x 256	*732*	*462*	*395 (Time sec.)*
	~75% efficiency		

As can be seen the advanced method outperforms that of coarse parallelism by several factors. Further improvement of the algorithm might yield an even greater efficiency.

3.2.5 Reverse - time migration impulse response

In order to examine the accuracy of the difference operator we need to test the depth migration response following the same steps as for the ω-x case, due to a time impulse as shown in Figure 3.5a. As expected, the operator approximates the semicircle closely because no square-root approximations were made to the differential equation apart from those corresponding to the derivatives. Therefore, the method should be reliable for up to 90^0 dips.

The time impulse in this test is at 0.3sec. with a homogenous velocity of 1000m/s for a model section with lateral spacing of 40m. The response of the fourth order difference approximation is shown in Figure 3.19. It can be seen by comparing with Figure 3.5a that the solutions approximate the semicircle response remarkably well.

3.2.6 Applications.

The reverse time migration algorithm was tested on the synthetic data over the step fault model Figure 3.6 which was also used for the ω-x code. As indicated above, the method can only be applied to limited data sets due to stability conditions and spacing constraints. The prestack reverse time migrated shot records are shown in Figure 3.20. It is clear that the method images the reflectors properly at greater offsets than the ω-x case (Figure 3.8) because of the higher degree (90^0) of accuracy. The final prestack migrated stack Figure 3.21 reveals the fault model properly, but due to spacing limitations in the code there is greater amount of noise present than the ω-x case.

No other data set was tested with this method. An improvement in the difference scheme could be investigated so that real seismic applications can be tested. The code was primarily developed in this work to investigate the degree of parallelism in finite-difference codes as been developed for seismic applications. In Chapter 5 the same techniques of parallelism are used to solve the three-dimensional acoustic wave equation.

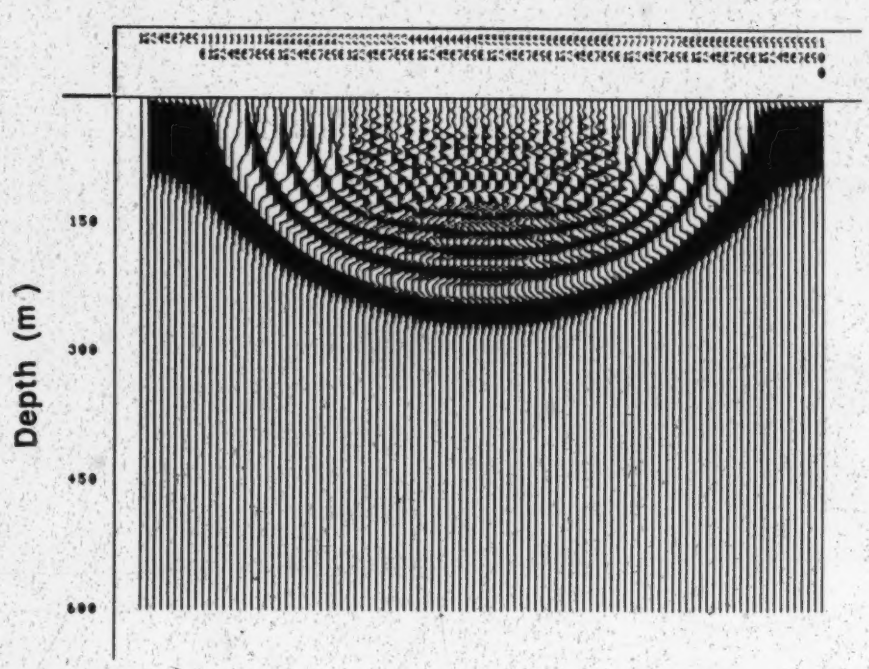


Figure 3.19. Impulse responses of fourth order reverse time migration.

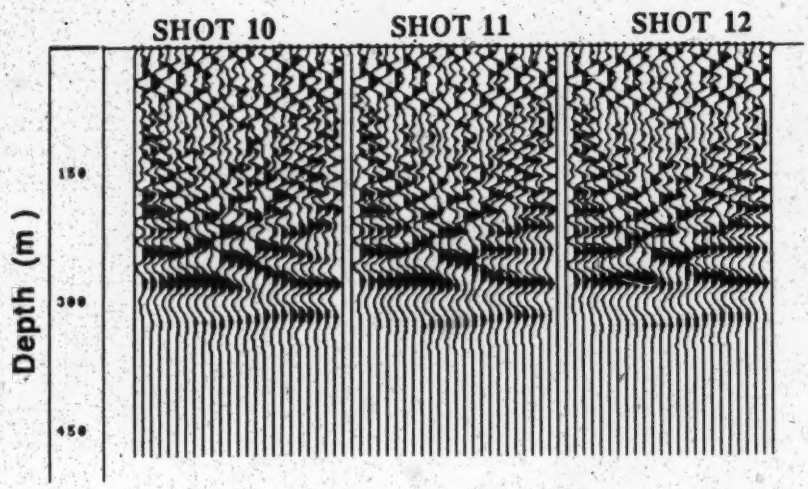


Figure 3.20 Prestack reverse time migrated shots.

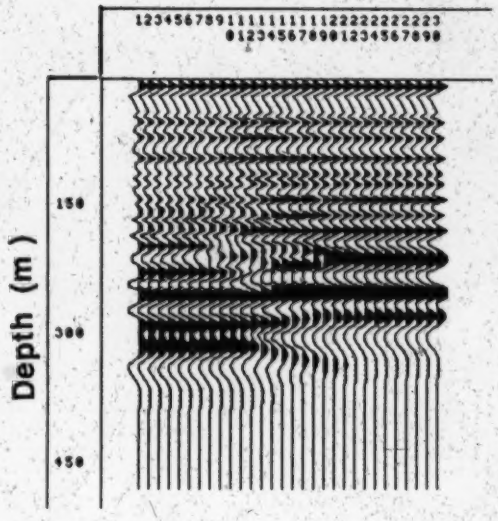


Figure 3.21 Prestack reverse time migrated stack of the step fault model.

CHAPTER 4

ONE PASS 3-D DEPTH MIGRATION WITH UNDER-RELAXATION

4.1 3-D POST STACK DEPTH MIGRATION

4.1.1 Introduction

The future of seismic processing in 3-D will require algorithm design for supercomputers. The general principles of such methods will be suitable for most types of parallel computers that can meet the challenge of dealing with the large 3-D data volume.

Subsurface geological features in hydrocarbon exploration are three dimensional in nature. For example salt diapirs, reefs, overthrust and folded belts, major unconformities and deltaic sands fall in this category. The 2-D seismic section is only a cross section of the 3-D seismic wavefield which contains signals from all possible directions. Therefore, this side energy can only be handled with 3-D processing or 3-D migration. The complex overburden problem has this three-dimensional character; hence, it is more accurate than 2-D. In recent years we have seen an increase in the use of 3-D migration methods in the poststack domain by means of two-pass and one-pass or splitting methods.

Two-pass migration, a method that replaces the true 3-D case with an approximation that solves a series of 2-D migrations in orthogonal directions (Brown, 1983; Dickinson, 1988, Yilmaz, 1987), has been used almost exclusively but is now virtually extinct. Possibly the reason for this is that splitting methods can be accommodated in the memory environment. This avoids incorrect migration of events that arise from the method even if a correct velocity model is used.

In some operational environments where the complete 3-D volume can be

accommodated the so called one pass or *alternating direction implicit* (ADI) 3-D approach to migration can be used.

The one-pass ADI method has been analyzed by several researchers and applied to the solution of the 3-D seismic problem (Yilmaz, 1987; Gibson et al. 1983; Jacubowicz and Levin, 1983). The method follows the classical solution of the system of equations, where for each depth step (or extrapolation step) the field is downward continued alternatively along the x and then y directions. This particular application, although a good approximation to the true solution, tends to introduce larger errors in the azimuthal direction of the wavefield.

Various authors have attempted to improve this solution by the introduction of extra calculations or other operators in the solution of the system of equations. Ristow (1980) and Cary (1990) used a symmetric approach to eliminate the anisotropic behaviour of the solution by migrating not only along x and y alternately but also along 45^0 to x and y. Hale (1990) has demonstrated another approach using McClellan transformations which are circular symmetric filters to eliminate the non-symmetric effect of ADI. Black and Leong (1987) used a partitioning approach to the downward extrapolator which avoids the splitting of the operator. Brown (1983) discussed the limitations of the operator separation approach under different velocity variations. Schneider (1978) showed an integral formulation of the 3-D migration problem solved by summation assuming no lateral velocity variations in the model. Stoffa et al. (1990) has shown a new dip splitting approach in the frequency domain. In terms of parallel one pass 3-D migration, Hague and Goloway (1989) have given an implementation of the w-x-y method of a 2 vector processor for the IBM 3090 environment.

In this chapter I will examine the parallel implementation of the ω-x-y migration on the SPS platform and also show a new approach that uses *under-relaxation* to solve the direct one-pass 3-D migration without splitting or separation. The extension of the method to

pre-stack domain will be also shown in the second part of this Chapter.

4.1.2 The method

One-pass 3-D migration of seismic data in the spatial-frequency domain, which includes lateral velocity variations typically requires more I/O operations and more computations than frequency-wavenumber algorithms. Many frequency-wavenumber algorithms are two-pass, involving 2-D migration in the x-direction followed by another 2-D in the y -direction.

In this Chapter, I will examine 3-D migration using an implicit method in the spatial-frequency domain with the algorithm that is based on the one-way 15 degree wave equation (which in practice is good for dips up to 30-35 degrees). The 45 degree solution also can be formulated as will be shown in Chapter 7 where I have solved the direct 3-D 45 degree paraxial wave equation.

Computer implementation of the above method uses the splitting approach or an ADI Crank-Nicolson scheme. Each downward continuation or extrapolation step proceeds along x and then y coordinates solving an *under-relaxed* tridiagonal system of equations. For each extrapolation the ADI method requires the solution of 2N complex tridiagonal systems of dimension N, where N is the number of input/output x and y coordinates. The algorithm is recursive where the result of the previous extrapolation step is required as an input to the current extrapolation step. Any laterally slowly varying velocity function can be used for the extrapolation.

To understand 3-D migration consider a point scatterer that is buried in a constant velocity medium. The zero offset traveltime curve in two dimensions is a hyperbola. We can imagine that the response in three dimensions is a hyperboloid. Migration in two dimensions, as shown in Chapter 3, amounts to summing amplitudes along the diffraction

hyperbola, then placing the result at the apex of it. This idea can be extended in three dimensions. Migration in this case amounts to summing amplitudes over the surface of the hyperboloid.

The formulation of the differential solution of the scalar wave equation as shown in the previous chapter in two-dimensions will be extended here to three space dimensions. The 15 degree approximation to the differential equation is used here but the 45 degree case can be implemented easily. The application of the method in the poststack domain with synthetic and real examples will be shown initially, with an extension to pre-stack 3-D examined at the end of this Chapter.

4.1.3 Application description

The observed seismic wavefield at z=0 can be approximated by the following 3-D scalar wave equation:

$$\frac{\partial^2}{\partial x^2} P(x,y,z,t) + \frac{\partial^2}{\partial y^2} P(x,y,z,t) + \frac{\partial^2}{\partial z^2} P(x,y,z,t) = \frac{1}{v^2(x,y,z)} \frac{\partial^2}{\partial t^2} P(x,y,z,t) \qquad (4.1)$$

Here, x, y and z are the spatial and depth coordinates and $v(x,y,z)$ is the velocity of the model medium.

The Fourier transformed field is given by:

$$P(k_x, k_y, z, \omega) = \int\int\int P(x,y,z,t) e^{(ik_x x + ik_y y - i\omega t)} dx\, dy\, dt \qquad (4.2)$$

Applying the operator from equation (4.2) to the scalar wave equation one gets get:

$$\frac{\partial^2 P}{\partial z^2} + \left\{ \left[\frac{\omega^2}{v^2(x,y,z)} - \left(k_x^2\right) \right] + \left[\frac{\omega^2}{v^2(x,y,z)} - \left(k_y^2\right) \right] \right\} P = 0 \qquad (4.3)$$

If multiple reflections are not prominent the solution to this equation may be obtained from the one-way scalar wave equation:

$$\frac{\partial P}{\partial z} = -i \frac{\omega}{v} \left[\sqrt{1 - \left(\frac{v^2 k_x^2}{\omega^2} \right)} + \sqrt{1 - \left(\frac{v^2 k_y^2}{\omega^2} \right)} \right] P \qquad (4.4)$$

with $v=v(x,y,z)$

Let us define:

$$k_z(z) = \pm \frac{\omega}{v} \left[\sqrt{1 - \left(\frac{v^2 k_x^2}{\omega^2} \right)} + \sqrt{1 - \left(\frac{v^2 k_y^2}{\omega^2} \right)} \right] \qquad (4.5)$$

where k_x, k_y and k_z are the two horizontal and vertical wavenumbers, and ω is the angular frequency. The choice of the minus sign in the square-root for migration is due to decreasing of traveltime as the field is moved downwards. Based on this conclusion equation (4.5) can be solved at the same depth step along x and y. Also a reformulation to incorporate lateral velocity variations is needed and requires that one moves from the wavenumber domain to spatial domain.

4.1.4 Lateral velocity variation

If lateral velocity variation is present $v=v(x,y,z)$ we need to bring equation (4.5) into

the spatial domain. To do this we need to evaluate efficiently and accurately the square root of a differential operator. The square root was approximated by a continued fractions expansion in Chapter 3 but here I am going to use a second order Taylor series expansion, leading to a 15^0 approximation:

$$k_z(z) = -\left(\frac{\omega}{v} - \frac{v k_x^2}{2\omega} - \frac{v k_y^2}{2\omega}\right) \tag{4.6}$$

and upon substitution

$$\frac{\partial P}{\partial z} = -i\left(\frac{\omega}{v(x,y,z)} - \frac{vk_x^2}{2\omega} - \frac{vk_y^2}{2\omega}\right)P \tag{4.7}$$

This is now a parabolic equation representing upgoing waves from reflectors to receivers. Claerbout (1985) showed that employing a retarded coordinate transformation removes the effect of vertical translation (see Chapter 3). This retarded coordinate system is related through the local velocity, v(z) at each source location to the Cartesian system by the following set of equations in 3-D:

$$\begin{aligned} t' &= t'(x,y,z,t) = t - \int_0^z dz/v(z) \\ x' &= x'(x,y,z,t) = x \\ y' &= y'(x,y,z,t) = y \\ z' &= z'(x,y,z,t) = z \end{aligned} \tag{4.8}$$

We can now write the pressure wavefield in terms of a stationary wave $Q(x,y,z,\omega)$ in the frequency domain as:

$$P(x,y,z,\omega) = Q(x,y,z,\omega)\ e^{-i\,\omega\int_0^z \frac{dz}{v(z)}} \qquad (4.9a)$$

or

$$Q(x,y,z,\omega) = P(x,y,z,\omega)\ e^{+\,i\,\omega\int_0^z \frac{dz}{v(z)}} \qquad (4.9b)$$

Differentiating (4.9a) with respect to z gives:

$$\frac{\partial P(x,y,z,\omega)}{\partial z} = \left(\frac{\partial}{\partial z} - \frac{i\omega}{v(z)}\right) Q(x,y,z,\omega)\, e^{\,i\,\omega\int_0^z \frac{dz}{v(z)}} \qquad (4.10)$$

Substituting equations (4.9a) and (4.10) into equation (4.6) we get the time shifted downgoing wavefield:

$$\frac{\partial Q(x,y,z,\omega)}{\partial z} = i\left(\frac{vk_x^2}{2\omega} + \frac{vk_y^2}{2\omega}\right) Q(x,y,z,\omega) - i\,\omega\left(\frac{1}{v(x,y,z)} - \frac{1}{v(z)}\right) Q(x,y,z,\omega) \qquad (4.11)$$

The first term on the right hand side represents the 3-D *diffraction term.* The second term on the right is the *thin lens* term. In the derivation above we assume that $v(x,y,z)$ is a locally constant medium velocity and $v(z)$ depends on depth below the source, only. Upon splitting the *diffraction term* becomes:

$$\frac{\partial Q(x,y,z,\omega)}{\partial z} = i\left(\frac{vk_x^2}{2\omega} + \frac{vk_y^2}{2\omega}\right) Q(x,y,z,\omega) \qquad (4.12)$$

Equation 4.12 may be rearranged by substituting for:

$-k_x^2 = \partial^2/\partial x^2$, $-k_y^2 = \partial^2/\partial y^2$ and $\mu(x,y,z) = \omega^2 v(x,y,z)$

Then the *diffraction* portion of the solution is given by the following approximation:

$$-2i\,\mu(x,y,z)\frac{\partial Q(x,y,z,\omega)}{\partial z} + \frac{\partial^2}{\partial x^2}Q(x,y,z,\omega) + \frac{\partial^2}{\partial y^2}Q(x,y,z,\omega) = 0 \qquad (4.13)$$

This equation will be evaluated step by step, starting at the surface and moving alternatively to any depth, using a finite difference approximation. The thin lens term may be solved analytically and is given by:

$$\frac{\partial Q(x,y,z,\omega)}{\partial z} = i\,\omega\left(\frac{1}{v(x,y,z)} - \frac{1}{v(z)}\right)Q(x,y,z,\omega) \qquad (4.14)$$

Equation (4.14) is also called the 15^0 approximation to the scalar wave equation, since this form of the square root approximation to the dispersion relation yield reliable results when the angles of incidence of the rays normal to the wavefront are less than about 15^0 as mentioned previously. It is also called a paraxial equation and has all the properties of parabolic differential equations. It is the direct 3-D equation for ω-x-y migration.

Converting equation (4.13) to a finite difference form (Appendix C) and writing it in matrix form for each harmonic yields an independent set of pentadiagonal solutions for each harmonic component, ω_n (n=1,2..) of the form:

$$-\bar{A}q^{z+1}_{x+1,y} - \bar{A}q^{z+1}_{x,y+1} + \bar{B}q^{z+1}_{x,y} - \bar{A}q^{z+1}_{x-1,y} - \bar{A}q^{z+1}_{x,y-1} = -\bar{C}q^{z}_{x+1,y} - \bar{C}q^{z}_{x,y+1} + \bar{D}q^{z}_{x,y} - \bar{C}q^{z}_{x-1,y} - \bar{C}q^{z}_{x,y-1} \qquad (4.15)$$

In the same way as with the 2-D solution we can propagate the wavefield in terms of each frequency in parallel, solving for all necessary depth steps, and summing at the end. The stability of the method can also be analyzed as in Appendix D, and shown to be stable unconditionally. The migrated result is obtained by applying the imaging condition summing the $Q(x,y,z)$ field over frequencies:

$$Q(x,y,z,t=0) = \frac{1}{N} \sum_{\omega} Q(x,y,z,\omega) \tag{4.16}$$

The application of this ω-x-y algorithm for migration is described below.

A major problem with the solution of equation (4.15) in pentadiagonal form (direct solution) arises from the fact that the computational time is of the order of N^3 where N is the x or y dimension. This type of solution will be shown in Chapter 7 along with the extra computing effort needed to solve it. My aim here is to transform equation (4.15) into a form that requires $2N^2$ operations per depth step. In order to achieve this goal I must solve the equation using the ADI approach by mapping the solution into a tridiagonal form through *under-relaxation*. In the next section I will describe the steps taken to metamorphose the equation (4.15) into such a form.

4.1.5 Matrix Solution Procedure

The 3-D solution of the wave equation creates a large system of linear, algebraic finite-difference equations, and many solution techniques are available. Matrix inversion and other direct techniques (Chapter 7) are usually unacceptable because of the large number of equations involved as well as the matrix size; therefore, iterative methods are normally employed or some formulations might utilize a Gauss-Seidel algorithm. This is a point-

by-point method in which the equations are solved one at a time, passing from node point to node point throughout the wave field. New values of the variables are used as soon as they become available, and the complete wave field can be solved for each dependent variable before going to the next dependent variable.

This idea, as implemented in an efficient Tridiagonal Matrix Algorithm (TDMA), follows the form of equation 4.17

$$b_i q_i = a_i q_{i-1} + c_i q_{i+1} + d_i \qquad (4.17)$$

Equation (4.17) requires the values of the constant coefficients a_i, b_i, c_i, and d_i for i ranging from 2 to $n-1$, in addition to the boundary values of q_1 and q_n.

Two new coefficients, e_i and f_i, are determined through the following recurrence relationships:

$$e_i = \frac{c_i}{b_i - a_i e_{i-1}} \qquad (4.18)$$

$$f_i = \frac{d_i + a_i f_{i-1}}{b_i - a_i e_{i-1}} \qquad (4.19)$$

with the starting values of $e_1=0$ and $f_1=q_1$. The solution procedure calls for successively rewriting equation (4.17) as

$$q_i = e_i q_{i+1} + f_i \qquad (4.20)$$

The e_i and f_i terms are found beginning with i+2 and proceeding to $i+n-1$, where N is the number of points in the line being solved. This step is an LU decomposition and forward substitution. New values of q_i are found by backward elimination, starting with

$i+n-1$ and working backwards to $i+2$. This algorithm is a special form of Gaussian elimination without pivoting, and the procedure is only stable when the matrix possesses diagonal dominance. Additional details concerning this procedure can be found in several sources (e.g. Golub and Van Loan, 1989).

The TDMA can be implemented to solve equation (4.15) by considering the equations for all the control volumes along a grid line with last or best estimates for the values of Q along the neighboring grid lines, and hence constructing a tridiagonal equation set. In this manner, the first transverse can proceed along all grid lines in the x direction. Then, using this solution as best estimate, the next step is to proceed along the y grid line within the same depth step.

Equation (4.15) is written by setting first

$$RHS = -\overline{C}q^{z}_{x+1,y} - \overline{C}q^{z}_{x,y+1} + \overline{D}q^{z}_{x,y} - \overline{C}q^{z}_{x-1,y} - \overline{C}q^{z}_{x,y-1} \qquad (4.21)$$

as the source term in the equation, along the initial x-direction sweep as:

$$\overline{B}q^{z+1}_{x,y} = \overline{A}q^{z+1}_{x+1,y} + \overline{A}q^{z+1}_{x-1,y} + \left[\overline{A}q^{z+1}_{x,y+1} + \overline{A}q^{z+1}_{x,y-1} + RHS\right] \qquad (4.22)$$

For the y-direction sweep is written as:

$$\overline{B}q^{z+1}_{x,y} = \overline{A}q^{z+1}_{x,y+1} + \overline{A}q^{z+1}_{x,y-1} + \left[\overline{A}q^{z+1}_{x+1,y} + \overline{A}q^{z+1}_{x-1,y} + RHS\right] \qquad (4.23)$$

The terms in parenthesis in each of these equations are considered known so that the TDMA can be applied. Each sweep though the matrix updates the values of Qi. The relationship between equation (4.15) and equation (4.17) can now be clarified. In each of two tridiagonal systems, two (the terms in parentheses in equations 4.22 and 4.23) of the four directional difference coefficients are incorporated into the source term, di (RHS).

The B term is renamed to be bi, leaving the backward and forward coefficients in the sweep direction to be ai and ci. Equation (4.15) is thus transformed to equation (4.17). The latter equation may be transformed into typical tridiagonal format through simple algebra. Equation (4.24) represents this change, and equation (4.25) describes the resulting matrix.

$$-a_i q_{i-1} + b_i q_i - c_i q_{i+1} = d_i \tag{4.24}$$

$$\mathbf{M}q = d, \text{ where } \mathbf{M} = f(a,b,c) \tag{4.25}$$

Two significant improvements were implemented in the 3-D migration matrix solution technique. First, the TDMA algorithm was solved via ADI in parallel over the line sweeps throughout the entire plane. A second major change in the TDMA was suggested by Van Doormaal (1990). This involved the implementation of an acceleration technique. The technique can be simply described as an under-relaxation of Q update for the off-diagonal terms. First, $Q^{BE}_{x,y+1}$, a better estimate of $Q_{x,y+1}$,is defined as follows:

$$Q^{BE}_{x,y+1} = Q^{z}_{x,y+1} + \theta\left(Q^{z+1}_{x,y+1} - Q^{z}_{x,y+1}\right) \tag{4.26}$$

This expression is then approximated through the following simplification:

$$Q^{BE}_{x,y+1} \approx Q^{z}_{x,y+1} + \theta\left(Q^{z+1}_{x,y} - Q^{z}_{x,y}\right) \tag{4.27}$$

It can be noted that θ is an under-relaxation parameter that must be specified between zero and one. If θ is set to zero, this procedure reverts to the normal TDMA procedure described by equations (4.17) to (4.25). If θ equals one, Q will never be updated and the

variables will remain constant though each iteration.

These changes alter the equation previously described. Equation 4.22 becomes:

$$\left(\overline{B}-\theta\overline{A}\right)q_{x,y}^{z+1} = \overline{A}q_{x+1,y}^{z+1} + \overline{A}q_{x-1,y}^{z+1} + \left[\overline{A}\left(q_{x,y+1}^{z} - \theta q_{x,y}^{z}\right) + \overline{A}q_{x,y-1}^{z} + RHS\right] \quad (4.28)$$

It should be noted that the *y-1* term is not under-relaxed because it has been updated from the previous iteration.

The equations in the TDMA solution algorithm require reformulation. A new parameter is first calculated:

$$g_i = \theta\overline{A} \quad (4.29)$$

equations 4.18 and 4.19 are altered as follows:

$$e_i = \frac{c_i}{b_i - g_i - a_i e_{i-1}} \quad (4.30)$$

$$f_i = \frac{d_i - g_i q_{x,y} + a_i f_{i-1}}{b_i - a_i e_{i-1}} \quad (4.31)$$

The TDMA derivation provided throughout this chapter applies only to the x-directional sweeps. Similar equations were derived and implemented for the y-directional sweep. It should be noted that the TDMA is an iterative solution technique to solve the finite-difference equations, the coefficients of which are only tentative and require updating to account for the changes in the values of the variables.

4.1.6 Under-relaxation

Simulations of the finite-difference equations in the form of equation (4.15) generally

diverge unless the iterative corrections are under-relaxed. The conventional practice in such case is to under-relax the dependent variables to avoid numerical instabilities. However, based on some tests, under-relaxation in this code is best accomplished when placed in the source terms. If α is the under-relaxation factor, than the two source terms in equation (4.15) become:

$$\overline{B} = \frac{\overline{B}}{\alpha} \tag{4.32}$$

and

$$RHS = RHS + \frac{(1-\alpha)Q^z_{x,y}\overline{B}}{\alpha} \tag{4.33}$$

where $Q^z_{x,y}$ is the known value of Q computed during the previous iteration.

These modifications are performed before the solution of the finite-difference equations by TDMA. It may also be required to under-relax the pressure wave properties if they experience rapid changes as in the case of sharp velocity contrast. If Q^{z-1} is the value of the property used in the previous extrapolation, n-1, and Q^z is the value computed in the present extrapolation, the values used in the next extrapolation step , Q are computed by :

$$Q = \alpha Q^z + (1 - \alpha)Q^{z-1} \tag{4.34}$$

The values of the under-relaxed factors should be between zero and one and for computational efficiency should be the largest value allowing a stable solution. The optimum values of these under-relaxation factors are problem dependent. Should divergence occur, it was found empirically that smaller values may be required.

4.1.7 Parallel implementation

The restructuring of this migration code in parallel was carried out similarly to the 2-D case. Based on the linear superposition of monochromatic harmonics in wave propagation, one may compute each frequency component of $Q(x,y,z,\omega_i)$ separately during steps described by equation (4.14) and (4.15). Keeping these facts in mind, the algorithm was designed as follows:

1. Temporally transform the wavefield Q by calculating the Fourier transform of all seismic records in parallel.
2. For each frequency step in parallel
2.1 for all depth steps
A. for all x directional sweeps (in parallel if more processors are available)
a. Estimate the time retarded field Q using (4.9).
b. For each depth step solve (4.15)

B. for all y-directional sweeps (in parallel if more processors are available)
c. repeat steps (a) and (b) until all depth points are resolved.
3. Automatic summing takes place over all depth steps in parallel for all the frequency components to get the final migrated section.

A description of the pseudocode is shown in Figure 4.1. The parallel steps over x and y directions can be invoked only if the number of processors exceeds the number of frequencies to be solved.

In terms of performance evaluation the code was running effectively (more than 90% user time) for all the examples tested in this work. A typical seismic survey consisting of 100 x 100 grid surface points extrapolated in depth to 1000m (500 steps) required a total time of 780s to be executed on the SPS-2 and 90s on the SPS-3. Analytical performances will be discussed during the examination of the applications of the algorithm.

ω-X-Y ONE PASS 3-D MIGRATION WITH UNDER-RELAXATION

(MAIN PART OF PARALLELISM)

(PARALLEL) DO over Frequency steps

DO over Depth steps

(PARALLEL) DO over number of lines(y)

A. Calculate the time retarded field

$$Q(x,y,z,\omega) = P(x,y,z,\omega)\ e^{\,i\,\omega \int_0^z \frac{dz}{v(z)}}$$

B. Downward continue the wave field using the diffraction term along x-direction using U.R.

$$-2i\mu(x,y,z)\frac{\partial Q(x,y,z,\omega)}{\partial z} + \frac{\partial^2}{\partial x^2} Q(x,y,z,\omega) + \frac{\partial^2}{\partial y^2} Q(x,y,z,\omega) = 0$$

END DO over lines (y)

(PARALLEL) DO over x - direction
REPEAT (B)
END DO over x

C. thin lens term

$$\frac{\partial Q(x,z,\omega)}{\partial z} = -\,i\,\omega\left(\frac{1}{v(x,z)} - \frac{1}{v(z)}\right) Q(x,z,\omega)$$

END DO LOOP over depth

END PARDO over frequencies

Figure 4.1 Pseudocode of the 3-D under-relaxed migration

4.1.8 3-D migration response

In order to evaluate the accuracy of the under-relaxed 3-D migration code the impulse response due to a point diffractor at a specific depth was calculated using the same model - only in 3-D here - as applied in the 2-D case (Chapter 3). The ideal 3-D impulse response is a hollow hemisphere. Although the traveltimes are not quite correct, the 15-degree response appears to have circular symmetry with respect to the center point.

Figure 4.2a shows the desired response in a 2-D slice of the cube. The under-relaxed response is shown in Figure 4.2b and the exact 3-D response calculated using the 15-degree version of the algorithm in Chapter 7 is shown in Figure 4.2c. We can see the level of approximation of the method to the correct response and also realize that it approximates the exact 3-D response reasonably well. Therefore, we can assume that the under-relaxed method gives a satisfactory solution to the 3-D problem at the 15-degree accuracy level. Figure 4.3a shows a depth slice (plane) of the under-relaxed 3-D as compared to the exact 3-D (figure 4.3b).

Again, the degree of approximation and the principle of under-relaxed solution seems to be a good alternative to the computation of the exact 3-D solution. The values for under-relaxation used for this example were θ=0.55 and α=0.65.

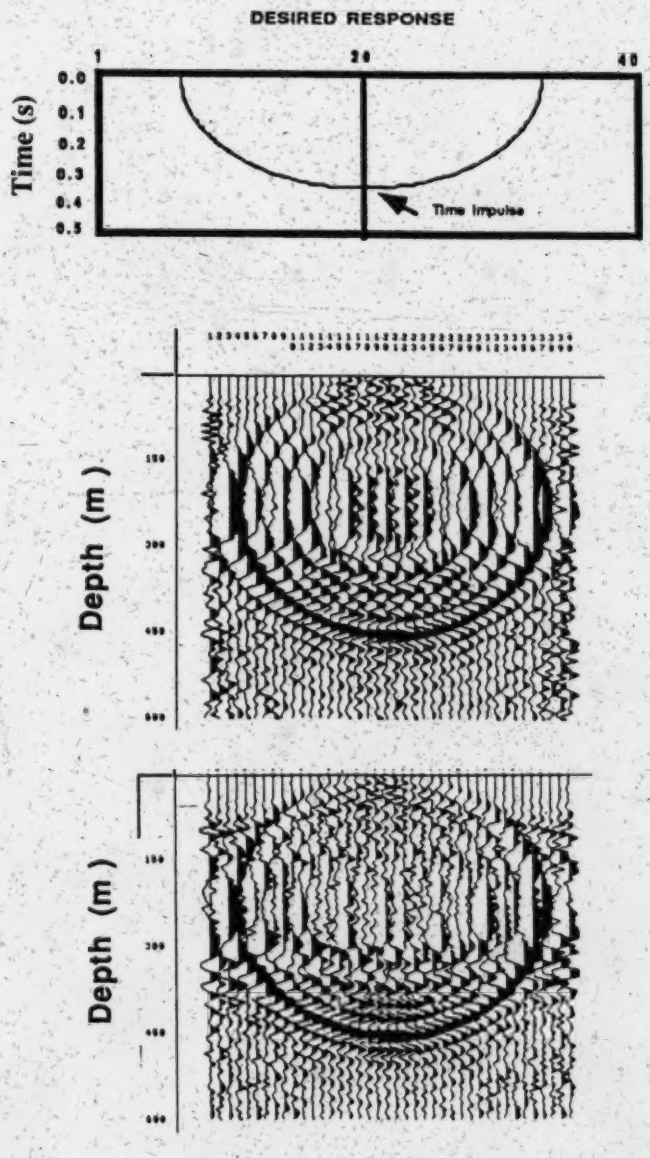


Figure 4.2 (a) Desired impulse response for migration, (b) 15^0 3D under-relaxed, (c) 15^0 3D exact response.

(a)

(b)

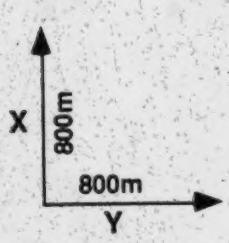


Figure 4.3 (a) Depth slice of the 3D under-relaxed migration response, (b) the 15 degree direct 3D equivalent.

4.1.9 APPLICATIONS OF THE 3-D POST-STACK MIGRATION

4.1.9.1 Simulated 3-D tank model seismograms

I have obtained a simulated 3-D tank model seismic data set from the University of Calgary through the courtesy of Professor Don Lawton. The tank model is a physical modeling lab system that can generate seismic data with almost any recording geometry over physical models of interest. The data generated here are from a zero-offset 3-D survey over a modified *French Model*. The data set comprised of 12,600 traces (105 lines in one direction x 120 in the other) over a 25 m x 25 m grid (scaled to field units). The reefs and fault blocks were made from plexiglas (v_p=2740 m/s) over an aluminum 'platform' (v_p=6200 m/s). Figure 4.4 shows a sketch of the plan view and cross sections show the geometry of the model.

The data were recorded with fixed gain with a sampling interval of 1ms, and no AGC (automatic gain control) has been applied to the display. A section showing a 2-D slice of this data is shown in Figure 4.5 where the reflections from the reef and the faults on the sides are evident. As can be seen, the data contain a significant amount of scattered energy from discontinuities of the model as well as side reflections. Therefore, it is a good data set to evaluate the accuracy of the 3-D migration code.

Figure 4.6 shows the migrated part of line 35 (figure 4.5). The method images the reef reflection as well as the fault reflection. Also the reflector at the bottom of the platform has been imaged properly. Some discrepancies are present with the imaging of the slopes of the reef and the fault which have a dipping angle greater that 40 degrees. A planar view of the migrated set is shown in Figure 4.7. It is evident that both reefs and faults have been imaged and migrated properly. It can be seen that a 2-D depth slice of the data gives a better physical understanding of the accuracy and efficiency of the 3-D migration algorithm.

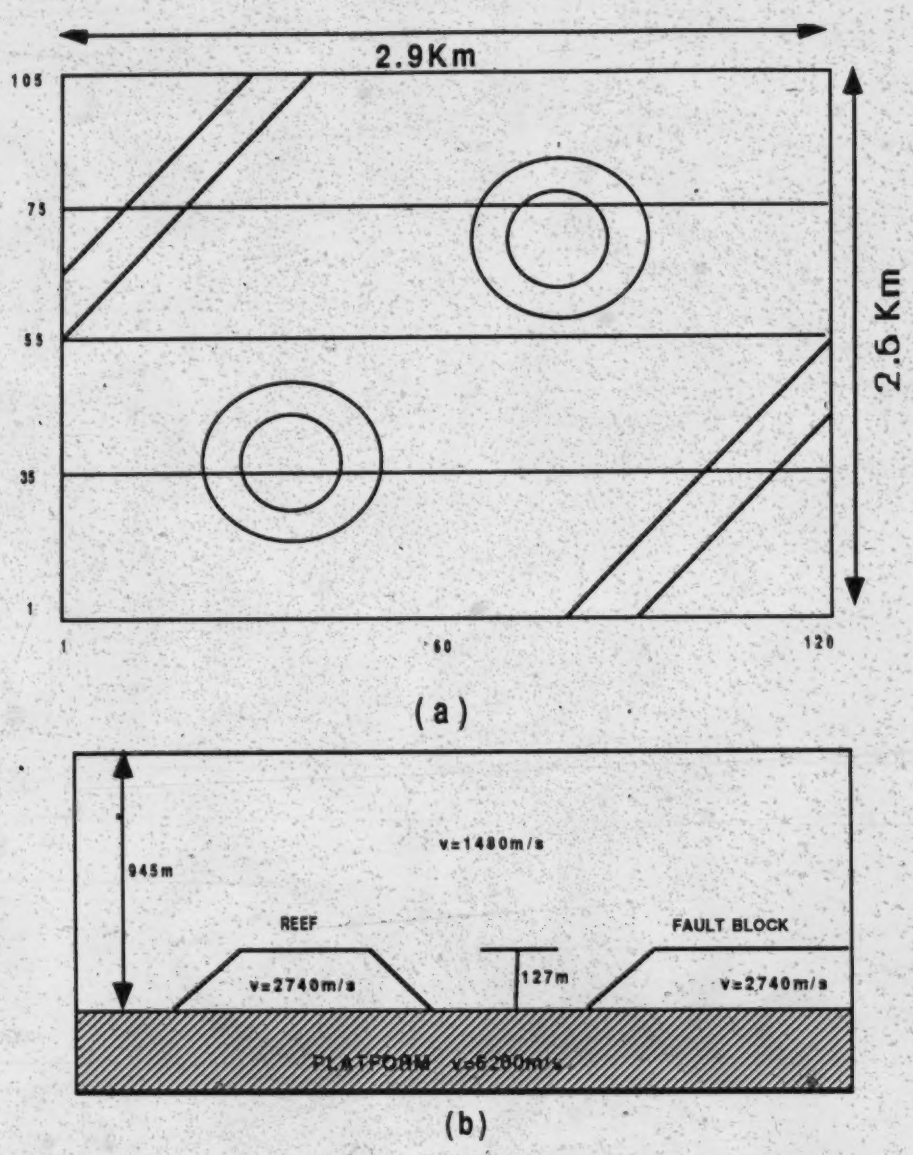


Figure 4.4 Diagrammatic sketch of (a) planar and (b) cross-sectional views of the 3-D tank model.

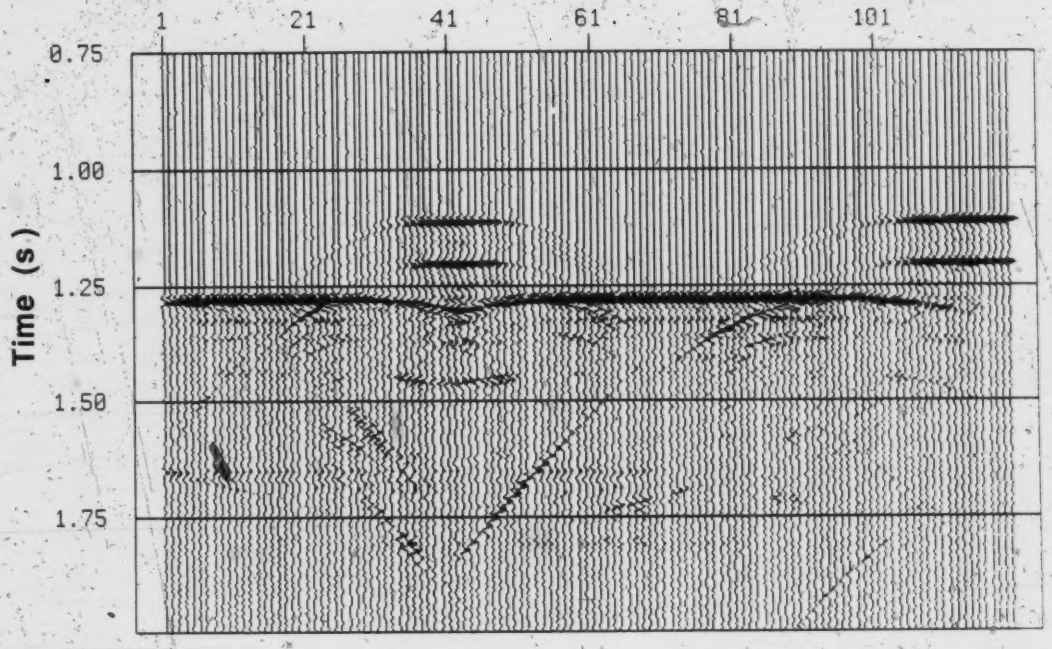


Figure 4.5 2D seismic section of line 35 from the tank model experiment.

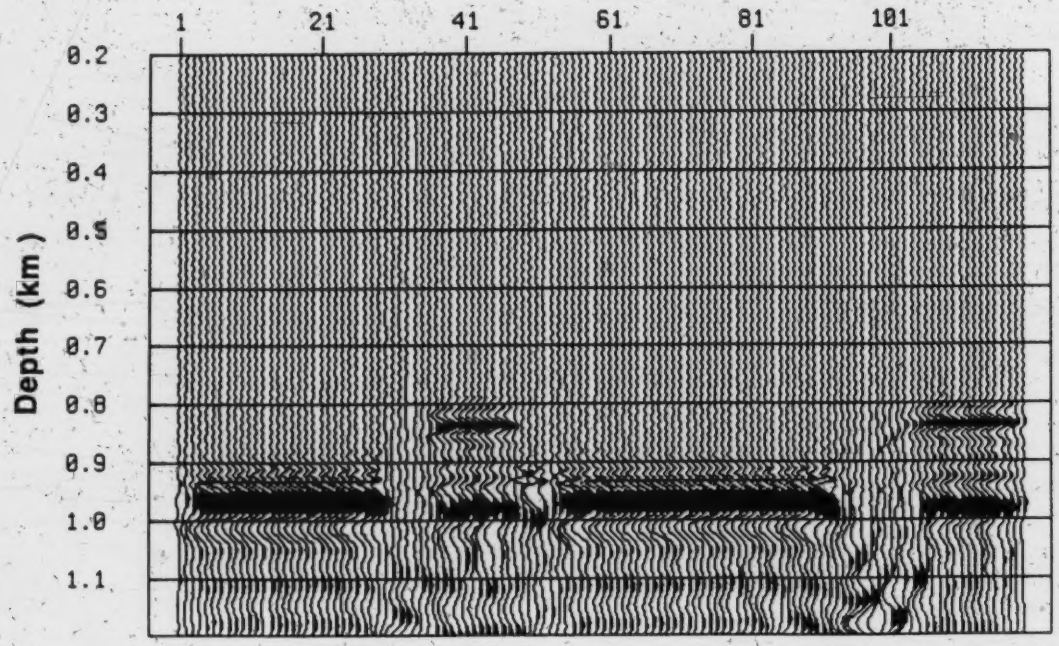
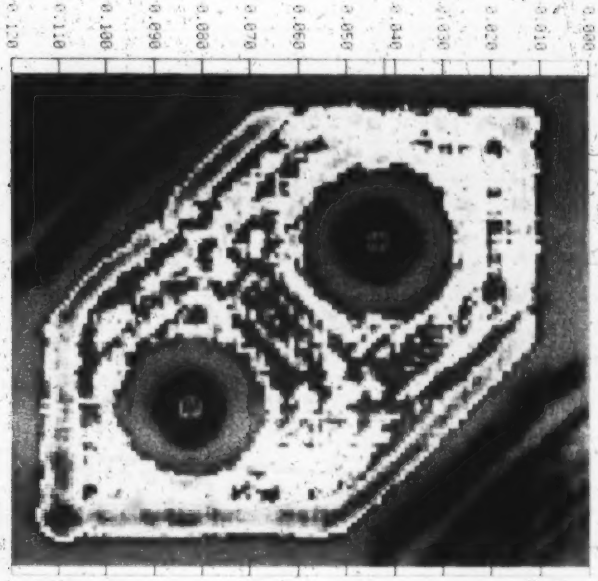


Figure 4.6 3D migrated plot of line 35. The positions of the reflectors form the top of the reef and fault (817m), the first interface at (945m) and the base of the model (1100m) have been imaged correctly. The sides of the reef and fault having a slope of 45^0 were not properly imaged



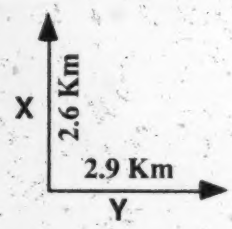


Figure 4.7 3D migrated depth slice of the tank model (830m). The reef and fault characteristics of the model are evident.

4.1.9.2. Imaging a 3-D zero-offset modeled data set.

The use of forward modeled zero offset stack data can be useful when evaluating the behaviour of a migration algorithm as well as give an intuitive picture of zero offset sections. The term zero offset section defines a seismic section generated under the assumption that data are obtained by coincident sources and receivers. The energy travel path from source to reflector is identical to that from reflector to receiver. This means that all sources are activated simultaneously, but each receiver records signals originating from the same source-receiver point. It should be clear that this approach does not simulate any wavefields resulting from experimental measurements.

The zero offset seismic section can also be modeled conceptually by a series of explosions detonated simultaneously at the reflected beds. According to the exploding reflector model (ERM), sources exist at every point on the surface of the reflectors. All these sources explode simultaneously at time t=0 with strengths proportional to their reflection coefficients and the wavefield travels upwards with a velocity equal to one half of the actual subsurface velocity. It is assumed that the wavefield recorded at the surface under these assumptions closely resembles a zero-offset section i.e. a stacked section.

For modeling start with a field $P(x,y,z=Z',\omega)=0$, where $z=Z'$ refers to a depth below all the reflectors. The wavefield is then extrapolated upwards in steps of Δz using equations for 3-D migration defined in Chapter 7. For this extrapolation the sign of Δz is negative since the wavefield moves in the negative z direction. At all those depths where sources exist an appropriate contribution to the wavefield is added for each frequency. The extrapolation is carried out till $z=0$ and then the resulting wavefield is inverse Fourier transformed with respect to ω to obtain a zero-offset section.

The method for migration and modeling described above is then tested for a specific 3-D model. Figure 4.8 illustrates a 30 x 30 x 80 model where there are two dipping layers, and a syncline between lines 10 to 20, and only two layers form line 1 to 9 and 21 to 30 as shown in Figure 4.8b. The velocities in each layer are also shown. For both modeling and migration we used a 20m trace interval and 4m depth step. A synthetic zero-offset section with 30x30 traces and 2 msec sampling interval was generated and part of it is shown in Figure 4.9. This forms the input for migration algorithm. Figures 4.10 and 4.11 illustrate the migration of the data by assuming all the layers have velocities equal to the vertical wave velocities. The top layer is properly imaged across all lines. The image of the graben is correctly imaged but appears to be present in lines before 9 and after 21. The reflector below is imaged correctly. For comparison the exact 3-D migrated data set is also shown here the graben appears to be properly imaged. It seems that the under-relaxed 3-D migration generated small error in imaging in the lateral direction as expected. In terms of real data situations such minor discrepancies may not be evident thus justifying the use of this type of migration.

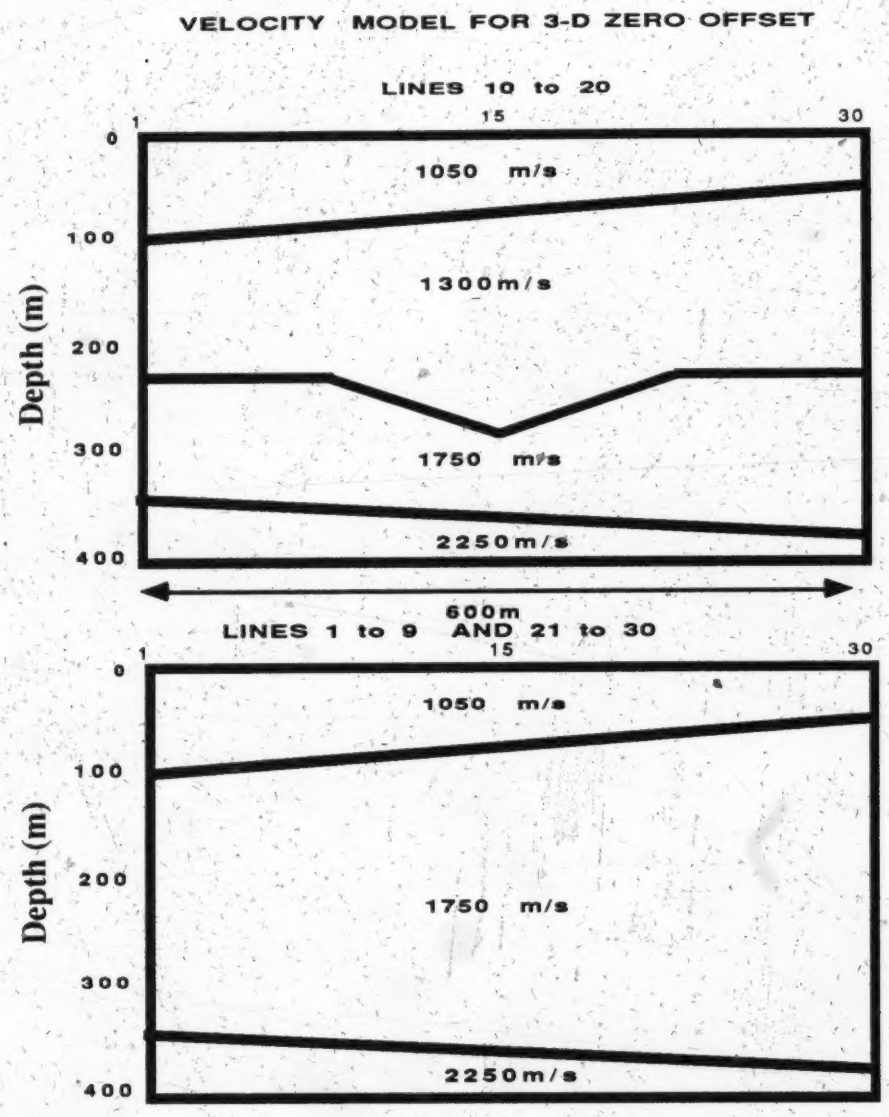


Figure 4.8 Diagrammatic representation of the physical model used for zero-offset modeling.

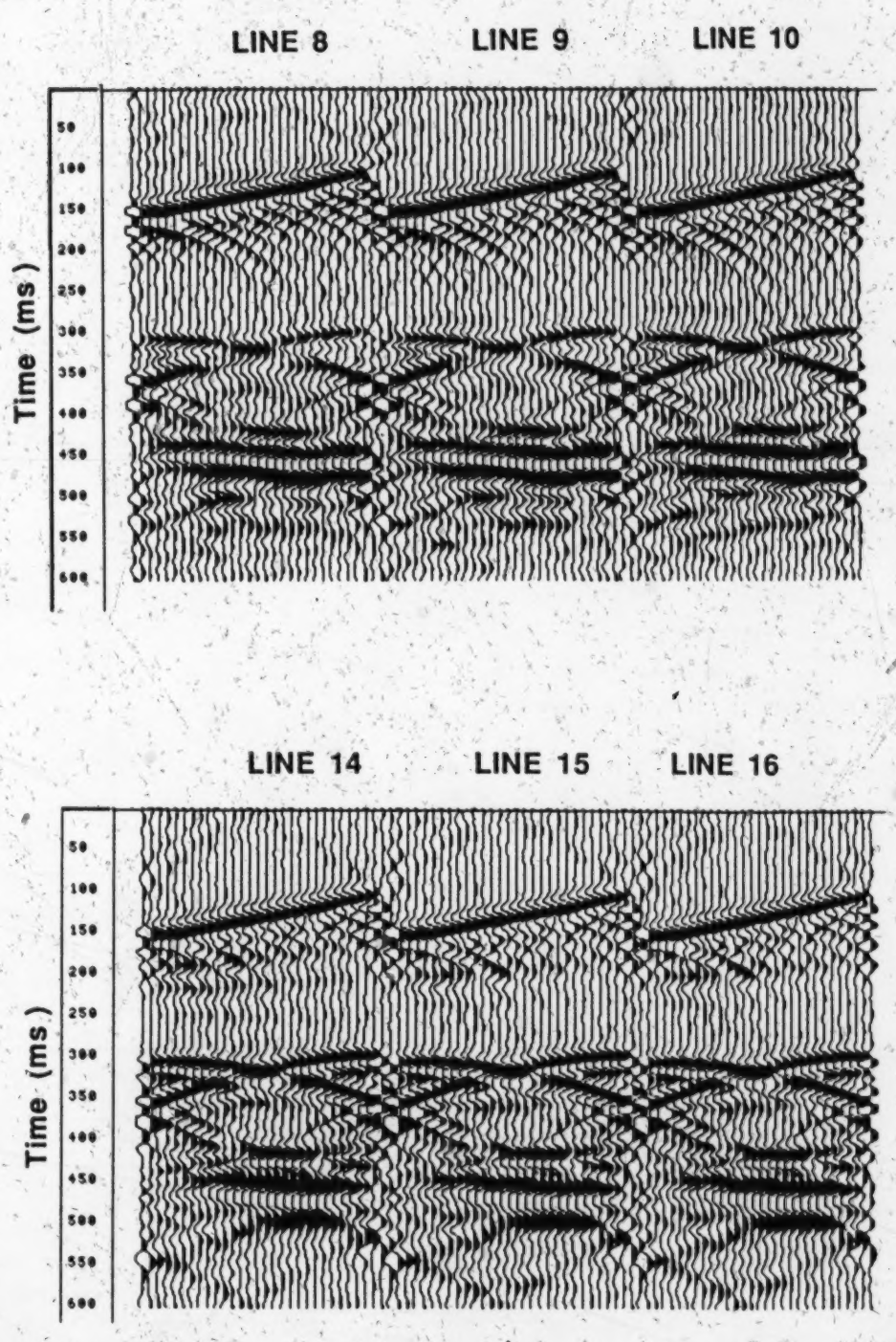


Figure 4.9 3D zero offset seismic response of the model shown in Figure 4.8

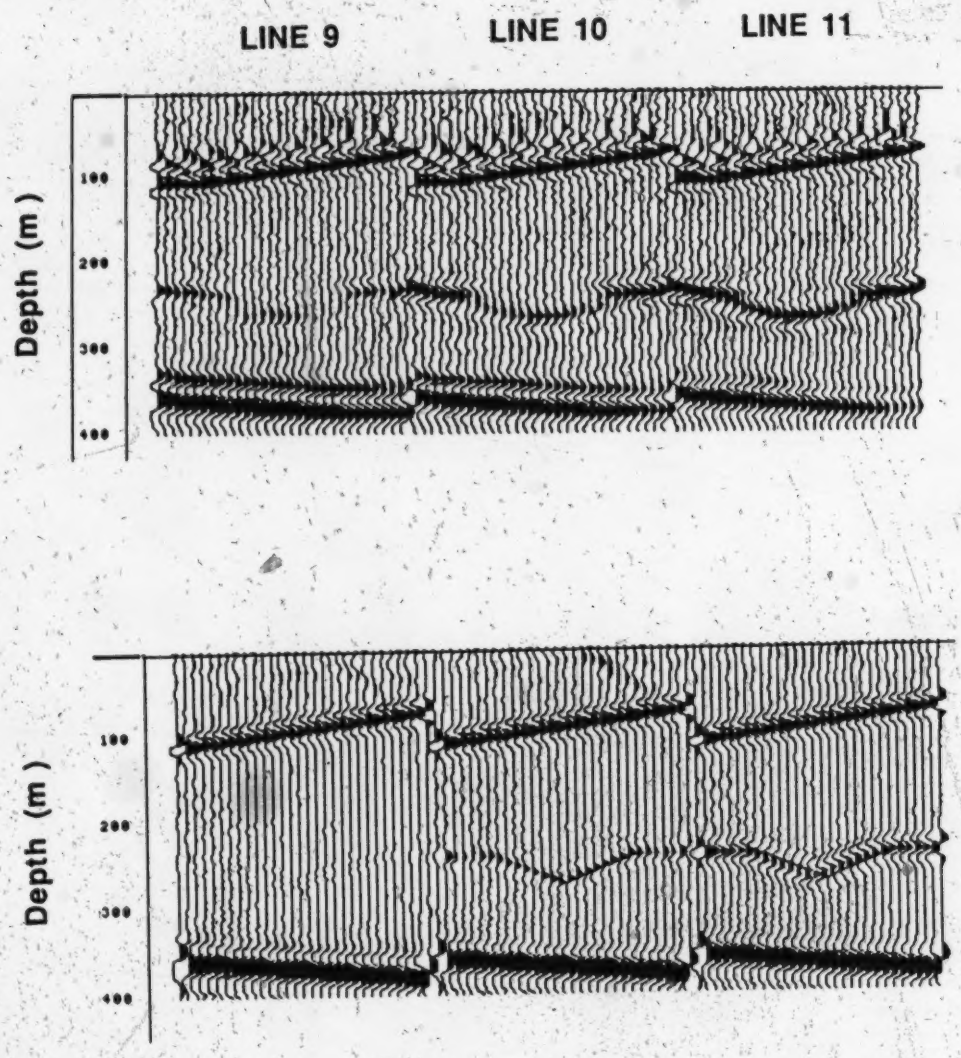


Figure 4.10 (a) 3D migrated sections via the under-relaxation method lines 9,10,11 (b) 3D migrated result using the exact 3D solution.

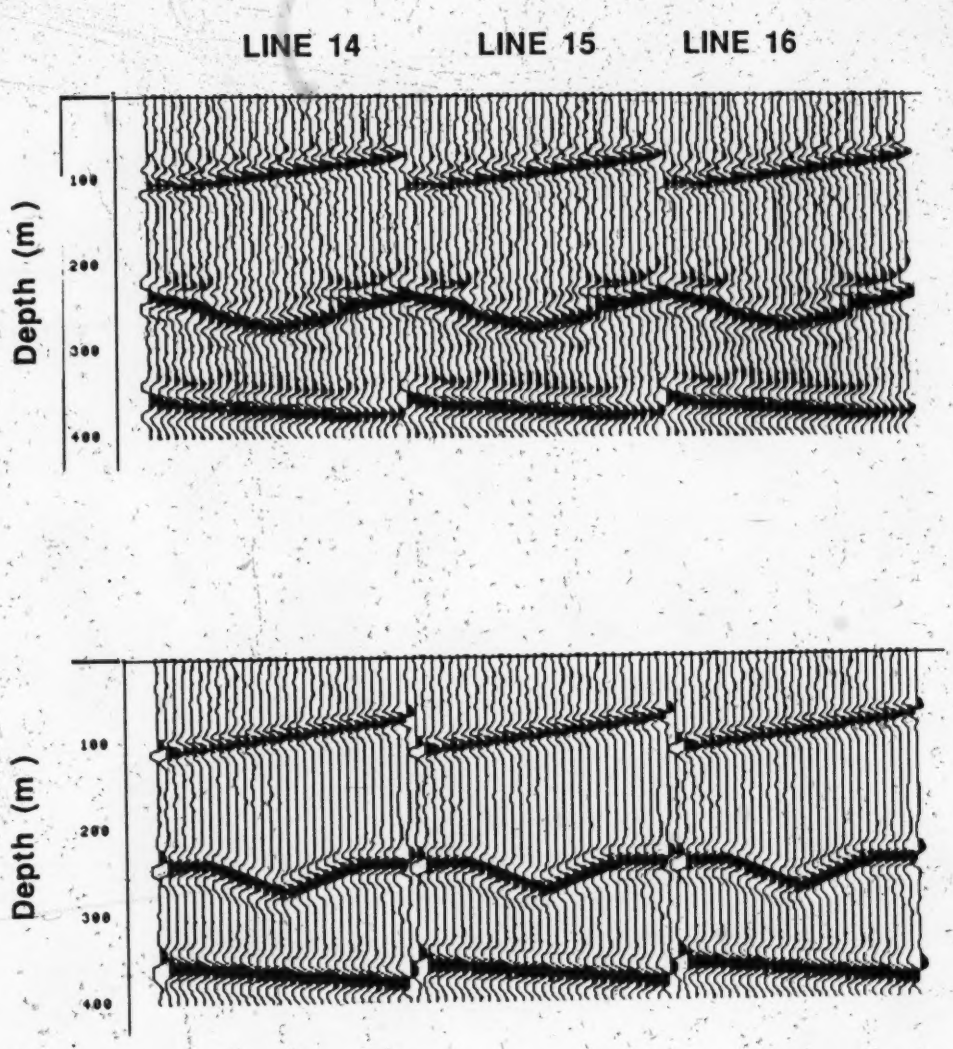


Figure 4.11 (a) 3D migrated sections via the under-relaxation method lines 14,15,16 (b) 3D migrated result using the exact 3D solution.

4.2 PRESTACK 3-D DEPTH MIGRATION VIA UNDER-RELAXATION

4.2.1 Introduction

As I have shown in Chapter 3, 2-D migration of seismic data before stack has its own merits. But taking into account the complexity of the geological environments investigated for hydrocarbon exploration 3-D prestack depth migration is desirable if one is willing to consider seriously the computational costs involved. As the real world is 3-D, most of the data should be migrated in 3-D as shown in the previous section. For these reasons as well as the availability of parallel computers I decided to extend the 3-D algorithm to the prestack domain.

Since 3-D prestack migration amounts, to an order of magnitude increase in the amount of information that must be processed as compared with the 2-D case, it is extremely important that the algorithm used for the migration be as efficient as possible. In particular, if the in-core memory of the data processing computer is of limited size, the major costs of a 3-D migration will typically result from the input and output of information to and from the permanent storage area. As this is not the case with most massively parallel computers the efficiency of the algorithm is expected to be good due to high locality (data reuse) of the operation.

3-D prestack depth migration research is very limited in the scientific literature. Kirchoff summation methods for 3-D prestack migration exist (Stolt and Benson, 1986) but they do not involve wave equation principles. Frequency-wavenumber methods are usually not feasible because they are limited to non-lateral velocity variations.

Recently some work has been done in this area (Froidevaux, 1990; Julien, 1990) where a 3-D prestack shot record migration is achieved in two steps: first, modeling of the propagation of the downward wave from the source, and second, back-propagation of the upgoing wavefield recorded at the receivers into the earth to a particular depth.

Berkhout (1982) and Claerbout (1985) give brief descriptions of the need and methodology

of 3-D prestack migration, without any mathematical formulation which can be effectively applied using today's supercomputers.

In this section I have developed a formulation for 3-D prestack migrations based on the same idea as for the 2-D case (Chapter 3). A double downward continuation for both source and receiver gather can be utilized effectively to accurately solve the problem. This method is considered more accurate than those described above because it is only constrained by a good velocity input model. The symmetric behaviour of source and receiver gather needed can be easily achieved by interpolation along the x or y directions.

4.2.2 Method description

Prestack migration in three dimensions refers here to the depth migration procedure applied to shot and receiver gathers. Before going into the mathematical implications of the problem (as seen in section 4.1), I would like to mention several important steps that need to be considered when dealing with a prestack 3-D seismic data space.

First, the potential of developing accurate 3-D velocity models is most of the time a difficult process, and because 3-D static corrections are important they must be done before depth migration. Moreover, the enormous amount of data requires great computational power. Keeping these considerations in mind, I have taken the steps of transforming the algorithm developed in the previous section for 3-D post stack migration to the prestack domain.

To remind ourselves, the idea of prestack depth migration conceptually involves shots and geophones that are separately downward continued into the earth, and the migrated depth section is picked off at zero travel time and zero offset for each z-level. In general, all of the recorded data must be worked with at once. First, common shot gathers are downward continued (using ADI and under-relaxation along x and y) by $\Delta z/2$. Then the

data are reorganized into common receiver gathers, and they are downward continued (x and y) by another Δz/2. In the space-frequency domain, this procedure can be done as I showed in Chapter 3 for all frequencies ω at the same time in parallel, and then a summation over ω will yield the imaged section. Due to the excess amount of data in 3-D, the option of running the code with a finer parallelism over shot and receiver gather is considered.

In section 4.1.2 the observed wave field at the earth surface in 3-D is approximated by equation (4.1). The source-receiver seismic wavefield at z=0 (earth's surface) can be approximated by a similar scalar wave equation in 3-D:

$$\frac{\partial^2}{\partial x^2} P(s_{xy},z,t) + \frac{\partial^2}{\partial y^2} P(s_{xy},z,t) + \frac{\partial^2}{\partial z^2} P(s_{xy},z,t) = \frac{1}{v^2(s_{xy},z)} \frac{\partial^2}{\partial t^2} P(s_{xy},z,t) \quad (4.35)$$

Here, s_{xy} and z are the spatial source and depth coordinates and $v(s_{xy},z)$ is the velocity of the model medium.

Equation 4.35 can be solved using the same approach as for equation 4.1 except that upon separation (Chapter 3) of shot and receiver spaces we end up with two *diffraction terms to* solve for:

$$-2i\,\mu(x,y,z) \frac{\partial Q(s_{xy},z,\omega)}{\partial z} + \frac{\partial^2}{\partial s_x^2} Q(s_{xy},z,\omega) + \frac{\partial^2}{\partial s_y^2} Q(s_{xy},z,\omega) = 0 \quad (4.36a)$$

where s_x and s_y are the shot gather x and y direction derivatives, and

$$-2i\,\mu(x,y,z) \frac{\partial Q(g_{xy},z,\omega)}{\partial z} + \frac{\partial^2}{\partial g_x^2} Q(g_{xy},z,\omega) + \frac{\partial^2}{\partial g_y^2} Q(g_{xy},z,\omega) = 0 \quad (4.36b)$$

where g_{xy} are receiver coordinates and g_x and g_y are the receiver gather x and y

direction derivatives.

Equations (4.36a) and (4.36b) are solved with the 15 degree under-relaxation method described in section 4.1. The restructuring of this migration code in parallel was carried out similarly with the 3-D post stack case. So the algorithm design is as follows:

1. Temporally transform the wavefield Q by calculating the Fourier transform of all seismic records in parallel.
2. For each frequency step in parallel
For all depth steps
A. Do over all the shot gathers (parallel)
a.do sx directional sweeps (in parallel if more processors are available)
1. Estimate the time retarded field Q using (4.10).
2. For each depth step solve (4.36a)
b do sy-directional sweeps (in parallel if more processors are available)
1. repeat step (a2).
B. for all the receiver gathers (parallel)
a.for all gx directional sweeps (in parallel if more processors are available)
1. For each depth step solve (4.36b)
b for all gy-directional sweeps (in parallel if more processors are available)
1. repeat step (a1)
3. Automatic summing takes place over all depth steps in parallel for all the frequency components to get the final migrated section.

A description of the pseudocode is shown in Figure 4.12. The code ran utilizing 90% of system time for computation for all the examples tested in this work. A typical seismic survey consisting of 121 shots on a 20 x 20 grid surface points extrapolated in depth to 400m (100 steps) required a total time of 18,000 s. to be executed on the SPS-2 and 1900s. on the SPS-3.

ω-X-Y PRESTACK ONE PASS 3-D MIGRATION WITH UNDER-RELAXATION

(MAIN PART OF PARALLELISM)

(PARALLEL) DO over Frequency steps

DO over Depth steps

A. (PARALLEL) DO over number of Shots

(PARALLEL) DO over number of Sx

1. Calculate the time retarded field

$$Q(x,y,z,\omega) = P(x,y,z,\omega)\ e^{\,i\,\omega \int_0^z \frac{dz}{v(z)}}$$

2. Downward continue the wave field along x using U.R.

$$-2i\mu(x,y,z)\frac{\partial Q(x,y,z,\omega)}{\partial z} + \frac{\partial^2}{\partial x^2} Q(x,y,z,\omega) + \frac{\partial^2}{\partial y^2} Q(x,y,z,\omega) = 0$$

END DO over Sx

(PARALLEL) DO over Sy

REPEAT (1)

END DO over Sy

B. REPEAT A over the receiver gathers.

C. Calculate the thin lens term

$$\frac{\partial Q(s_{x,y}z,\omega)}{\partial z} = -\,i\,\omega\left(\frac{1}{v(s_{x,y},z)} - \frac{1}{v(z)}\right) Q(s_{x,y},z,\omega)$$

END DO LOOP over depth

END PARDO over frequencies

Figure 4.12. Pseudocode of the prestack 3-D under-relaxed migration

4.2.3. Synthetic Data Application

In order to test the 3-D prestack algorithm a simulated seismic 3-D data set had to be developed. Real field data that fall into the category of equi-spaced grids were not available for this research. I will utilize a synthetic data set generated using a 3-D acoustic wave forward modeling algorithm developed by the author and described in the first part of Chapter 5.

The physical model used for the generation of the seismograms is shown in Figure 4.13. The planar view shows a step fault model extending across the y grid direction and the survey direction running such that 11 x 11 traces will be generated using this geometry. In this manner the central part of the model will be illuminated by the shot records. The source locations are shown on the diagram each one having 20x20 (400) receivers. The model data were generated using a $\Delta x=\Delta y=20$m, and Δz =10m spacing and time sampling of 1.5 ms.

Figure 4.14 shows a shot record generated by the forward modeling program. At the top a 2-D cross section view of the cube slices is shown with a planar view at 600ms shown in the bottom part. Clearly, the reflection from the edge of the fault is shown with the diffraction characteristics evident.

All these shot records (121 in total) were 3-D prestack depth migrated. The corresponding migrated shot from Figure 4.14 is shown in Figure 4.15. The top is the cross sectional view and bottom the planar view at 450m. From the planar view we can see that the method images correctly the reflector at that depth around the source position. As we move away from the source the accuracy of imaging deteriorates.

Finally, Figure 4.16 shows the 3-D prestack migrated stack of all the shot records. Two planar depth slices are shown: one at a depth of 450m and the other at 500m; at the top and bottom reflector positions in the fault. The method seems to image correctly the

edge of the fault as well as the position of the reflectors.

4.3 Conclusion

The first part of this chapter I have shown a new way of solving the 3-D migration problem using an *alternating direction implicit* method with *under-relaxation*. The method is based on the formulation of exact 3-D problem which avoids the drawbacks of operator splitting and achieves a reasonably accurate solution. Tests on synthetic and real (Chapter 7) examples will verify the fact that this approach of solving the 3-D methods has merits and is worth investigating further with a higher degree of approximation.

In the second part of this chapter, the analysis of 3-D prestack migration was shown based on the double downward continuation method of prestack imaging and solved through *under relaxation*. A synthetic example that justified the accuracy of the method was shown. Further improvements in the solution could improve the dynamic imaging capabilities of this procedure.

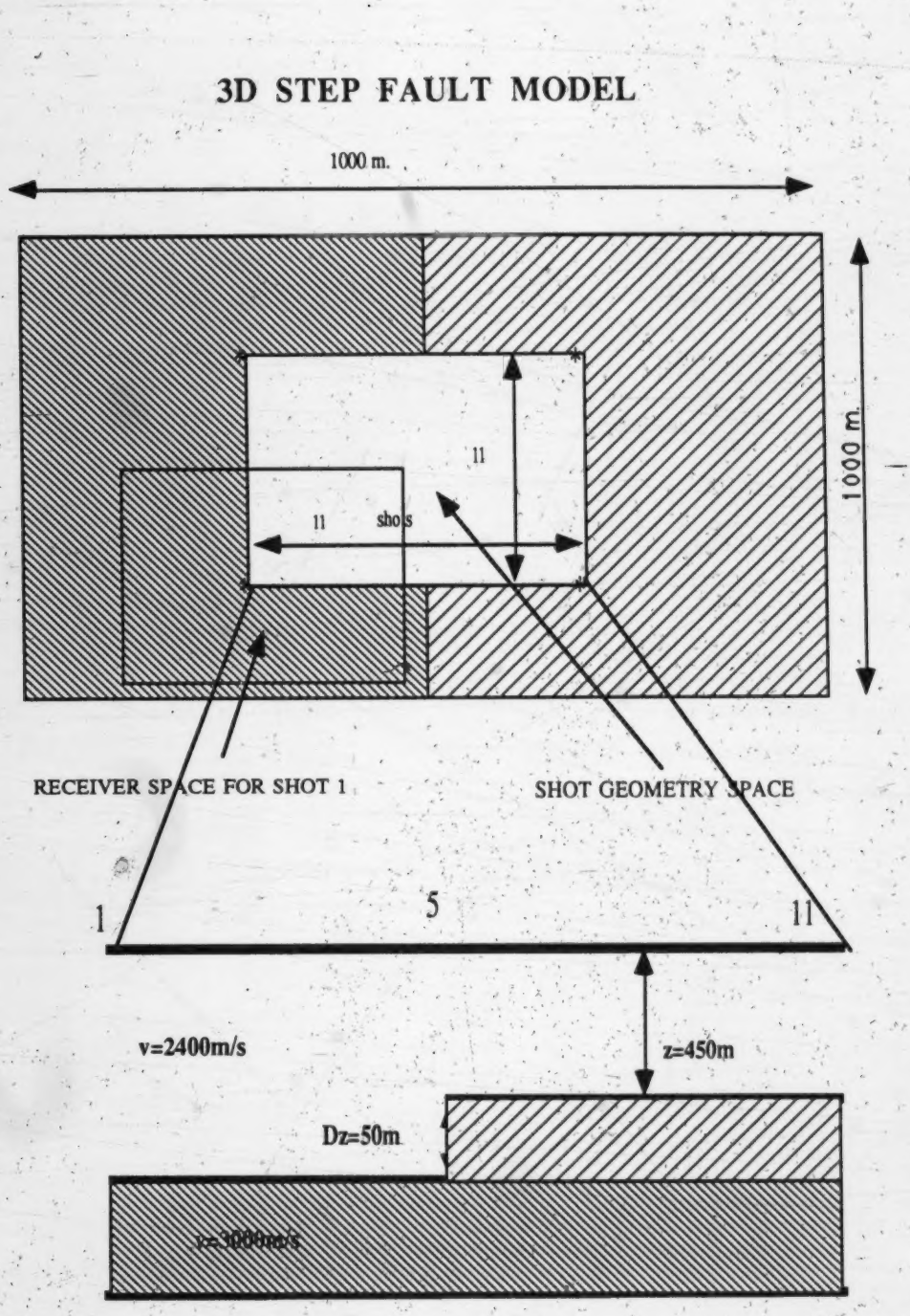


Figure 4.13 Schematic representation of the 3-D model used for the generation of synthetic seismograms.

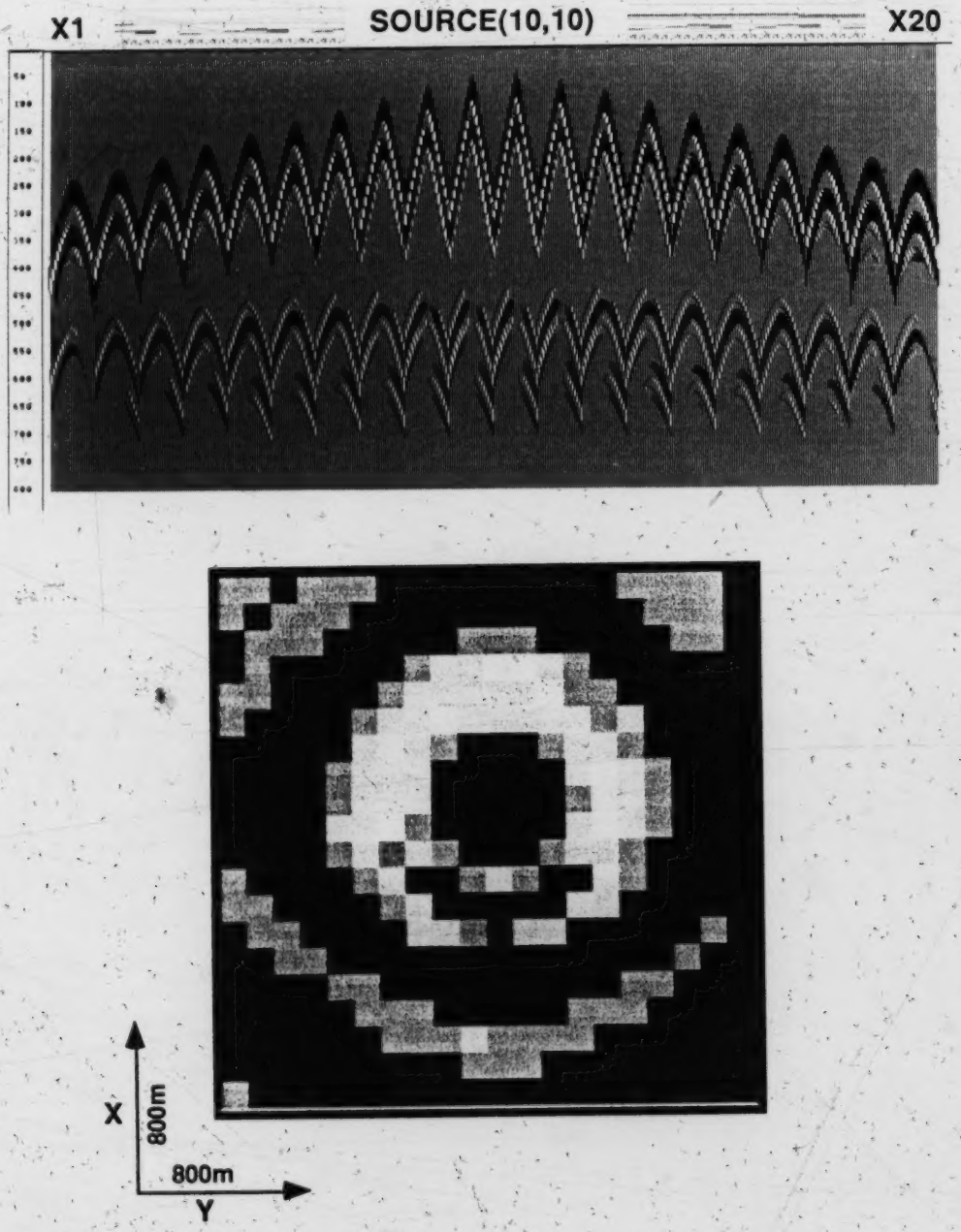


Figure 4.14 Shot record generated via 3-D forward modeling using the model in Figure 4.13. Top cross sectional view, bottom planar view at 600ms.

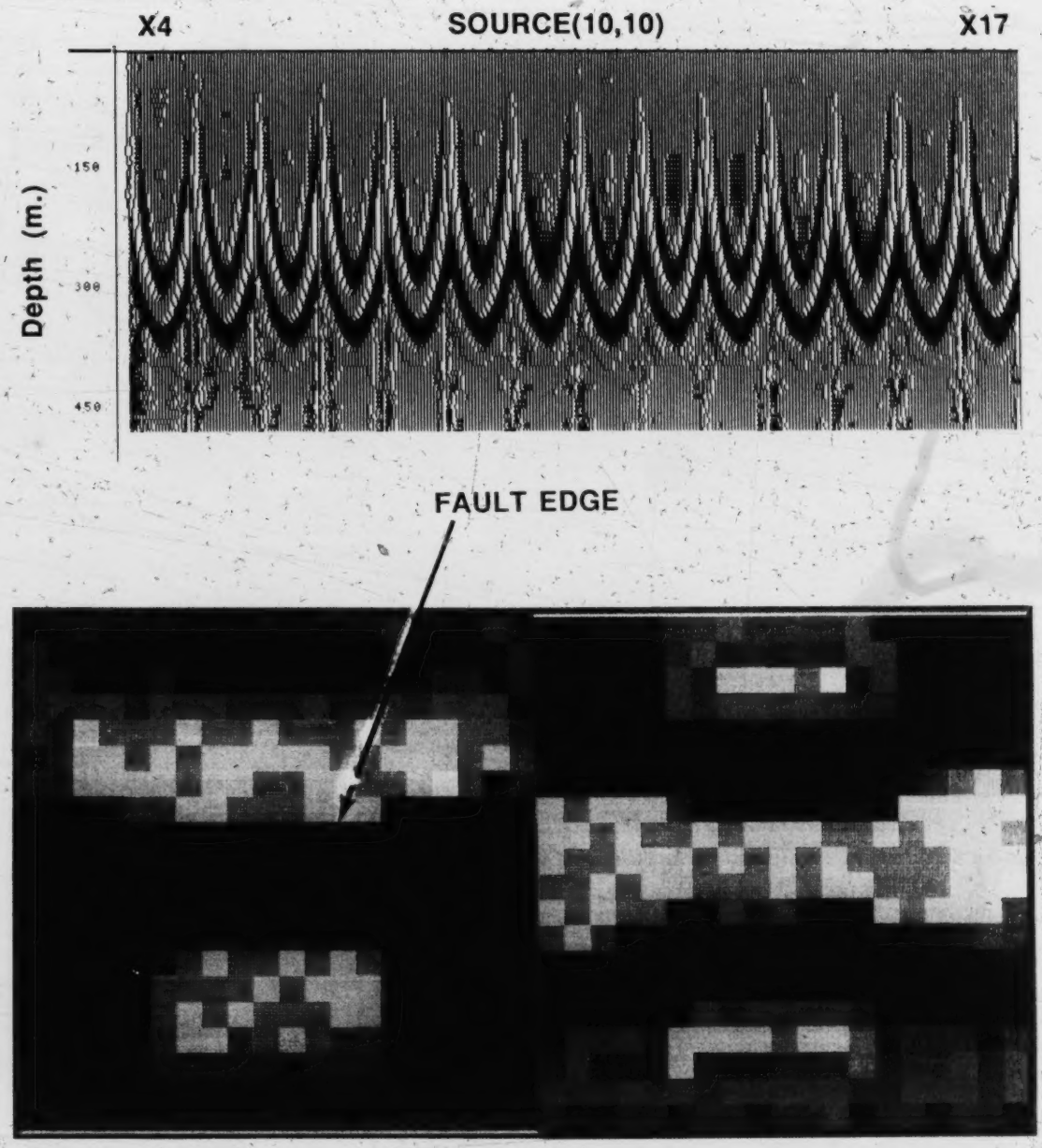


Figure 4.15 Migrated shot record of the data in Figure 4.14. Top cross sectional view, bottom planar views at 450m and 500m.

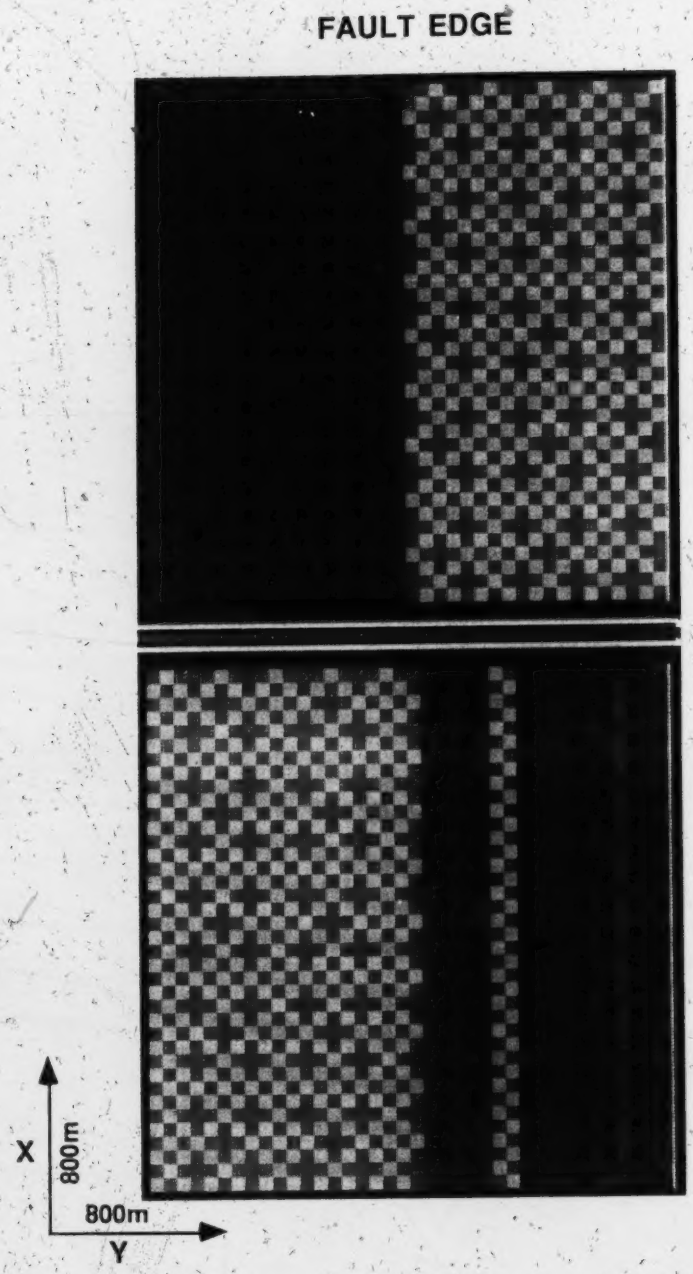


Figure 4.16 Prestack migrated stack section of the 3D step fault model. Depth slices shown are at depths of 450m and 500m., located at the top of the reflectors present in the model.

CHAPTER 5

3-D ACOUSTIC SEISMIC MODELING IN PARALLEL

5.1 Introduction

In order to fully investigate the seismic response associated with complex geological structures, several years ago, the geophysical community launched a series of efforts to develop 2-D finite-difference seismic models. It was a big step forward as compared to conventional modeling methods of ray tracing. Recently, with the introduction of more powerful supercomputers, it is feasible to investigate the modeling problem in three dimensions.

The advantages of 3-D forward seismic modeling are many. Some of the advantages include more dependable interpretation, better understanding of amplitude variations, more accurate velocity analysis and determination of data acquisition parameters. But there are also some drawbacks. Even on supercomputers, such models require an enormous amount of CPU time and a huge memory for data manipulation.

A plethora of research work related to the solution of the 2-D acoustic wave problem can be found in the literature but it is not my intention to review this in this thesis. There exist several methods for 3-D acoustic forward modeling and the basic mathematical problem formulations are well known. However, most of these cases utilized vector supercomputers, for example, Reshef et al. (1988) directly solved the acoustic wave equation using spatial derivatives in the Fourier domain on a CRAY X-MP. Jonson (1984) followed a time domain approach developed for the Cyber205. Mufti (1989) presented the formulation of the problem for the Cray 2 platform. Some work has been done in SIMD parallel computers (CM2) in 3-D (Mora, 1988) and 2-D (Myczkowski et

al., 1991) with impressive results, due to the fine-grained parallelism present in the algorithm. No actual implementation of the solution of the 3-D problem has been attempted for MIMD environments.

In this Chapter, I am going to present the formulation of the 3-D acoustic wave propagation problem and describe its implementation on an MIMD parallel computer. Although, this type of hardware requires coarse-grained parallelism present in a program, the large problem size and the computation of more than one shot records at the time contain enough work to saturate all processors, leading to efficiency and overall speedup. Based on this observation the MIMD environment could be a good production tool for several seismic records whereas the SIMD can only generate quickly one such record at a time.

5.2 The 3-D acoustic wave equation and its F-D solution

The treatment of the 3-D seismic modeling problem will be based on the solution of the acoustic wave equation using explicit finite-differences. Usually this is a good approximation for both marine and land models as long as we do not record three-component data and continue to use this equation at important steps during data processing.

Consider a 3-D space in which the z-axis, positive downward, denotes depth below the surface of the earth which coincides with the plane $(x,y,0)$. In this coordinate frame, the acoustic wave equation can be expressed as:

$$P_{xx} + P_{yy} + P_{zz} = \frac{1}{c^2} P_{tt} + f(t)\delta(x-x_s)\delta(y-y_s)\delta(z-z_s) \qquad (5.1)$$

where

$c(x,y,z)$=velocity of the medium

$P(x,y,z)$=pressure

$f(t)$=a time-dependent source located at (x_s,y_s,z_s)

The subscripts in equation (5.1) indicate derivatives of the wavefield with respect to x,y,z and t. For the purpose of setting up a finite-difference model, it would be convenient to introduce a set of indices i,j,k and n such that

$$\begin{aligned} x&=i\Delta x \\ y&=j\Delta y \\ z&=k\Delta z \\ t&=n\Delta t \qquad\qquad i,j,k,n=0,1,2... \end{aligned} \tag{5.2}$$

In equation (5.2), Δx, Δy and Δz denote uniform grid spacings along x, y and z axes respectively and Δt means the time sampling interval. By using these indices, we can write:

$$\begin{aligned} P(x,y,z,t) &= P^n_{i,j,k} \\ f(t) &= f_n \qquad\qquad n=0,1,2... \end{aligned} \tag{5.3}$$

$P^n_{i,j,k}$ denotes the discrete value of the wavefield at the grid point (i,j,k) at time n. A similar notation can be used to indicate discrete values of related quantities such as P_{xx}.

Using central differences (Smith, 1965) the first term on the LHS of equation (5.1) can be approximated as:

$$(P_{xx})^n_{i,j,k}=\frac{P^n_{i-1,j,k}-2P^n_{i,j,k}+P^n_{i+1,j,k}}{(\Delta x)^2}+O\left[(\Delta x)^2\right] \tag{5.4}$$

The remaining wavefield derivatives can be treated in a similar manner. For the source term we can write

$$f(t) = \begin{cases} f_n & \text{at } (i_s, j_s, k_s) \\ 0 & \text{elsewhere} \end{cases} \qquad n=1,2\ldots \tag{5.5}$$

Substituting expressions such as (5.4) and (5.5) into (5.1) we get,

$$\begin{aligned} P^{n+1}_{i,j,k} = {} & a_{i,j,k}\left(P^n_{i-1,j,k} - 2P^n_{i,j,k} + P^n_{i+1,j,k}\right) \\ & + b_{i,j,k}\left(P^n_{i,j-1,k} - 2P^n_{i,j,k} + P^n_{i,j+1,k}\right) \\ & + e_{i,j,k}\left(P^n_{i,j,k-1} - 2P^n_{i,j,k} + P^n_{i,j,k+1}\right) \\ & + 2P^n_{i,j,k} - P^{n-1}_{i,j,k} - (c\Delta t)^2 f_n \delta(i-i_s)\delta(j-j_s)\delta(k-k_s) \end{aligned} \tag{5.6}$$

where

$$\begin{aligned} a_{i,j,k} &= \left(c_{i,j,k}\frac{\Delta t}{\Delta x}\right)^2 \\ b_{i,j,k} &= \left(c_{i,j,k}\frac{\Delta t}{\Delta y}\right)^2 \\ e_{i,j,k} &= \left(c_{i,j,k}\frac{\Delta t}{\Delta z}\right)^2 \end{aligned} \tag{5.7}$$

It is usually feasible to set $\Delta x = \Delta y = \Delta z = h$

In that case (5.6) reduces to

$$\begin{aligned} P^{n+1}_{i,j,k} = g_{i,j,k}\left(P^n_{i-1,j,k} + P^n_{i+1,j,k} + P^n_{i,j-1,k} + P^n_{i,j+1,k} + P^n_{i,j,k-1} + P^n_{i,j,k+1} - 6P^n_{i,j,k}\right) \\ + 2P^n_{i,j,k} - P^{n-1}_{i,j,k} \end{aligned} \tag{5.8}$$

where

$$g_{i,j,k} = \left(c_{i,j,k} \frac{\Delta t}{\Delta h} \right)^2 \tag{5.9}$$

Equation 5.8 is a second order approximation of the wave equation in space and time. Higher order approximations can be derived by retaining more terms in the Taylor series expansion (5.4) to the accuracy of $O\left[(\Delta x)^4\right]$. The result can be expressed in the form

$$(P_{xx})^n_{i,j,k} = \left[I - \frac{\delta_x^2}{12} \right] \frac{\delta_x^2}{(\Delta x)^2} P^n_{i,j,k} + O\left[(\Delta x)^4\right] \tag{5.10}$$

where I is the identity matrix and

$$\delta_x^2 P^n_{i-1,j,k} = P^n_{i-1,j,k} - 2P^n_{i,j,k} + P^n_{i+1,j,k} \tag{5.11}$$

so the left hand side of the equation (5.1) can be expressed as:

$$(P_{xx})^n_{i,j,k} = \frac{1}{12(\Delta x)^2} \left(-P^n_{i-2,j,k} + 16P^n_{i-1,j,k} - 30P^n_{i,j,k} + 16P^n_{i+1,j,k} - P^n_{i+2,j,k} \right) + O\left[(\Delta x)^4\right] \tag{5.12}$$

Similar expressions can be derived for the other spatial derivatives.

Fourth order expressions for the spatial derivatives will be used, but only a second order expansion for the time derivatives. By doing so the resulting difference relation for the 3-D wave equation can be expressed as:

$$
\begin{aligned}
P^{n+1}_{i,j,k} = \frac{1}{12}\Big\{ & a_{i,j,k}\Big[P^n_{i-2,j,k} + P^n_{i+2,j,k} - 16\Big(P^n_{i-1,j,k} + P^n_{i+1,j,k}\Big) + 30P^n_{i,j,k}\Big] \\
& + b_{i,j,k}\Big[P^n_{i,j-2,k} + P^n_{i,j+2,k} - 16\Big(P^n_{i,j-1,k} + P^n_{i,j+1,k}\Big) + 30P^n_{i,j,k}\Big] \\
& + c_{i,j,k}\Big[P^n_{i,j,k-2} + P^n_{i,j,k+2} - 16\Big(P^n_{i,j,k-1} + P^n_{i,j,k+1}\Big) + 30P^n_{i,j,k}\Big]\Big\} \\
& + 2P^n_{i,j,k} - P^{n-1}_{i,j,k}
\end{aligned}
\tag{5.13}
$$

Finally, by setting $\Delta x=\Delta y=\Delta z=h$ the relation reduces to:

$$
\begin{aligned}
P^{n+1}_{i,j,k} = \bar{g}_{i,j,k}\Big[& P^n_{i-2,j,k} + P^n_{i+2,j,k} P^n_{i,j-2,k} + P^n_{i,j+2,k} + P^n_{i,j,k-2} \\
& + P^n_{i,j,k+2} - 16\,(P^n_{i-1,j,k} + P^n_{i+1,j,k} + P^n_{i,j-1,k} + P^n_{i,j+1,k} \\
& + P^n_{i,j,k-1} + P^n_{i,j,k+1}) + 90P^n_{i,j,k} \Big] + 2P^n_{i,j,k} - P^{n-1}_{i,j,k}
\end{aligned}
\tag{5.14}
$$

where

$$
\bar{g}_{i,j,k} = -\frac{\left(c_{i,j,k}\dfrac{\Delta t}{h}\right)^2}{12}
\tag{5.15}
$$

Relation (5.14) is only slightly more involved than relation (5.8) and yet it permits the use of much coarser grid intervals. Its simplicity makes it very powerful candidate for designing efficient 3-D models. I have used equation (5.14) for the purpose of modeling 3-D acoustic waves in heterogeneous media but the option to use relation (5.8) exists in the algorithm.

5.3. Dispersion and numerical stability

The maximum value of grid spacing which can be used in a model without causing excessive dispersion of energy is governed by the relation (Mufti, 1989):

$$max(\Delta x, \Delta y, \Delta z) = \frac{min(c_1, c_2, \cdots, c_m)}{w f_{max}} \qquad (5.16)$$

where

m = number of velocity types in the model

f_{max} = maximum frequency in the source wavelet

w = number of samples per wavelength corresponding to f_{max}

In equation (5.14) we may set $\Delta x = \Delta y = \Delta z = h$. In that case, this relation yields the largest value of h, say h_{max}, for a given model. However, if we choose different values of Δx, Δy, and Δz, each of these equations may be less but none of them may exceed h_{max}. Therefore, the use of equal grid spacings along all the three axes of coordinates minimizes the population of grid points and it represents the most efficient choice.

In equation (5.14), used for the formulation of the 3-D algorithm in this Chapter, dispersion problems arise only if w<3. Typical grid dimensions for a 3-D model based on equation (5.14) with w=3 are usually 250x250x250. This results in 15.625 million grid points. These numbers explain why the development of seismic models had to be restricted to two dimensions in the past. The use of algorithms such as that presented in equation (5.13) coupled with the availability of parallel supercomputers with memories beyond 256 Mbytes have opened the avenue for 3-D seismic modeling.

The problem of numerical stability can be considered as follows: The largest value of the time sampling interval which can be used in a given model without making the system numerically unstable modified from 2-D (Mitchell, 1969; Alford, 1974) to 3-D here is given by,

$$\frac{(\Delta t)_{max}\, max\,(c_1, c_2, \cdots, c_m)}{max\,(\Delta x, \Delta y, \Delta z)} \leq \mu \qquad (5.17a)$$

or

$$(\Delta t)_{max} \leq \mu \frac{h}{C_{max}} \qquad (5.17b)$$

The quantity μ depends on the algorithm used for computing the wavefield; it can be determined by following the von Neumann method (Smith, 1965). In the case of equation (5.14), $\mu = \sqrt{\frac{3}{8}}$ which is somewhat less than in the case of equation (5.8) which yields $\mu = \frac{1}{\sqrt{2}}$. Thus, the advantage of using larger values of grid spacing is somewhat offset by the more restricted choice of μ. In practical terms, it is not a severe constraint.

5.4 Absorbing boundary conditions

Relation (5.14) indicates that for computing $P^{n+1}_{i,j,k}$, we must know the quantity $P^{n}_{i+2,j,k}$. This implies that for computing the wavefield along the grid plane $i=I$, we must know the field for time n along the plane $i=I+2$ which lies outside the model. A similar situation arises along grid lines such as $i=I$ and $j=J$ which are adjacent to the boundaries of the model and it gets worse along the grid lines which are located at the intersection of such planes. This problem can be easily avoided by using asymmetric operators for evaluating the wavefield at such grid locations.

Another problem which is much more troublesome is the unwanted reflections from the subsurface boundaries of the model. These boundaries correspond to five different planes, $i=0$, $i=I+1$, $k=0$, $k=K+1$ and $j=J+1$. Thus, it is a much bigger problem than the corresponding problem which arises in 2-D models. Several approximate algorithms are available for getting rid of such reflections (Clayton and Engquist, 1977; Raynolds, 1978; Korn and Stoekel, 1982) all of which lead to more or less similar results. Raynold's algorithm seems to be the simplest to be converted to 3-D space. Generally, the method includes the elimination of the wavefield for the new time step (n+1) by using the

following relations along the boundaries.

$$\left.\begin{aligned} &\frac{\partial P}{\partial x} - \frac{1}{c}\frac{\partial P}{\partial t} = 0 \quad i=0 \\ &\frac{\partial P}{\partial x} - \frac{1}{c}\frac{\partial P}{\partial t} = 0 \quad i=I+1 \end{aligned}\right\} \quad j=0,\ldots,J+1;\ k=0,..,K+1 \tag{5.18}$$

$$\left.\begin{aligned} &\frac{\partial P}{\partial y} - \frac{1}{c}\frac{\partial P}{\partial t} = 0 \quad j=0 \\ &\frac{\partial P}{\partial y} - \frac{1}{c}\frac{\partial P}{\partial t} = 0 \quad j=J+1 \end{aligned}\right\} \quad i=0,\ldots,I+1;\ k=0,..,K+1 \tag{5.19}$$

$$\left.\begin{aligned} &\frac{\partial P}{\partial z} - \frac{1}{c}\frac{\partial P}{\partial t} = 0 \quad k=K+1 \end{aligned}\right\} \quad i=0,\ldots,I+1;\ j=0,..,J+1 \tag{5.20}$$

Derivation of difference relations for equations (5.18) to (5.20) involves first-order differencing.

As an example for equation (5.18) we get:

$$\begin{aligned} P^{n+1}_{0,j,k} &= P^{n}_{0,j,k} + \frac{C\Delta t}{\Delta x}\left(P^{n}_{1,j,k} - P^{n}_{0,j,k}\right) \\ P^{n+1}_{I+1,j,k} &= P^{n}_{I+1,j,k} - \frac{C\Delta t}{\Delta x}\left(P^{n}_{I+1,j,k} - P^{n}_{I,j,k}\right) \end{aligned} \quad j=0,\ldots,J+1;\ k=0,..,K+1 \tag{5.21}$$

Similar relations can be derived for equations (5.19) and (5.20). Raynold's method has shown effectiveness in reducing reflections from the boundaries in the test examples used during this research. But there is still some residual energy which escapes into the model. The approach of Israeli and Orszag (1981) can be applied that uses damping of the reflected signals but was not implemented in this algorithm.

Reflections from the surface were dealt with, with stress free boundary conditions that can be easily defined and applied.

5.5 . Solution and computer implementation

5.5.1 Method of solution

The solution procedure of equation (5.13) and the generation of a 3-D synthetic seismograms that I have adopted in the algorithm is as follows: Consider a simple 3-D model consisting of different material velocities. It extends along the x-axis from $i=0$ to $i=I+1$, along the y-axis from $j=0$ to $j=J+1$, and the z-axis from $k=0$ to $k=K+1$. The subsurface boundaries of this model are defined by the planes $i=0$ and $k=K+1$ and keep the size of the model finite. The value of the wavefield or its derivative is usually specified along the boundaries of the model. Then with the help of the initial conditions:

$$\begin{aligned} P^0_{i,j,k} &= 0 \\ (P_t)^0_{i,j,k} &= 0 \end{aligned} \tag{5.22}$$

the field is evaluated successively for $n=1,2,...$ at a set of grid points $\{(i,j,k)\}$.

The way that I initiated the evaluation of the wavefield was by setting:

$$P^1_{i,j,k} = \begin{cases} f_1 & at \ (i_S,j_S,k_S) \\ 0 & elsewhere \end{cases} \tag{5.23}$$

where f_1 is the first sample of a point source function (appropriate for 3-D modeling and described below) that is input at the source location (i_S,j_S,k_S). Looking at equation (5.13) we see that for $n=1$ all the values of the field in the left hand side are known; thus the value of the wavefield for time $n=2$ can be calculated for the various grid points. This process will be repeated until the maximum desired time step N has been completed. The

shot record information is stored in an array of data collected at $\left\{P^{n}_{i,j,0}\ (n=1,\ldots,N)\right\}$ for a fixed value of i or j.

5.5.2. The source function

As shown in equation (5.1), the function that generates the initial values of the modeled wavefield f(t) was called the source. Designing appropriate source functions for finite-difference modeling plays a very important role to the solution of the wave equation. There are generally two type of sources applied to these problems, point sources and line sources. Line sources are mainly used in 2-D modeling where the accuracy of a point source deteriorates. In 3-D modeling it is actually better to use a point source because the disturbance generated by it is naturally confined to propagate in 3-D space.

Source functions can be estimated by a number of mathematical representations (Aki and Richards, 1980). The majority of them have a time variation of a Gaussian function or its derivative. For this work I have chosen to use the second derivative of a Gaussian function whose length depends upon the dominant frequency used and the sampling rate required. That is:

$$G(t) = Ce^{-\sigma^2(t-t_1)^2} \tag{5.24}$$

with $\sigma=\pi f_0$ where f_0 is the dominant frequency of the source, t_1 is a time delay, and C can be determined after the application of the second derivative which gives:

$$G''(t) = 2\sigma^2\left(1 - 2\sigma^2(t-t_1)^2\right)e^{-\sigma^2(t-t_1)^2} \tag{5.25}$$

Figure 5.1 shows the type of Gaussian function used for source in the 3-D algorithm for f_0=30Hz and Δt=1ms and source length of 103 ms.

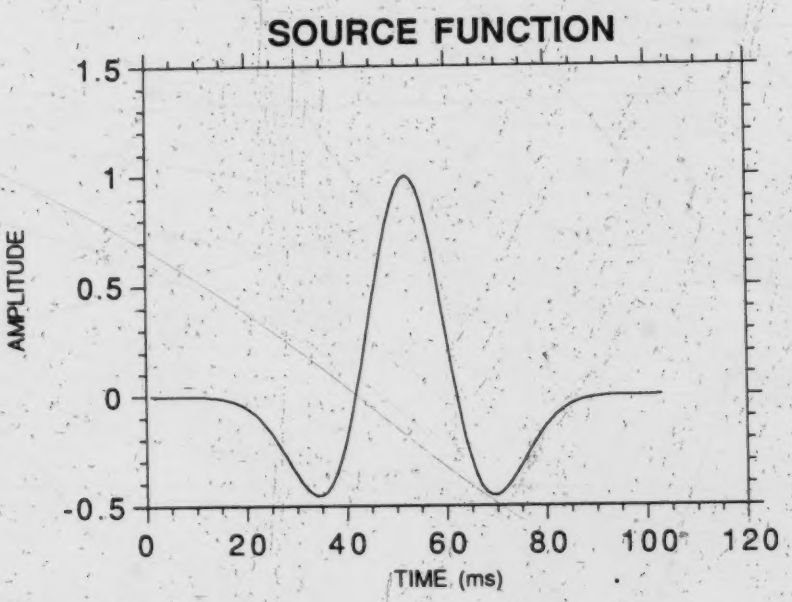


Figure 5.1 Gaussian source function for $f_0 = 30$Hz and $\Delta t = 1$ ms.

5.5.3 Parallel implementation and performance

Methods described in Chapter 2 for the solution of finite difference problems in parallel using the SPS computer hardware will be applied to the solution of the 3-D acoustic wave equation. Finite differences can be computationally expensive when the spatial resolution is high and/or the velocities of the model are quite large. The efficiency of the method depends greatly on the technique of calculating the wavefield amplitudes at each time step.

The advent of parallel computing opens new avenues in the applications of finite differences. In this section a description of algorithm formulation in terms of coarse parallelism over space is shown for the second order difference scheme for illustration purposes. The fourth order scheme can been analysed similarly. I did not attempt to apply partial time parallelism in this algorithm because the efficiency achieved by the coarse method was satisfactory.

In general the wavefield at an advanced time step is given by terms of second order differences looking in a one dimensional perspective:

$$\begin{bmatrix} P^1 \\ P^2 \\ P^3 \\ P^4 \end{bmatrix}^{t+1} = \begin{bmatrix} 2-2g & -g & 0 & 0 \\ -g & 2-2g & -g & 0 \\ 0 & -g & 2-2g & -g \\ 0 & 0 & -g & 2-2g \end{bmatrix} \begin{bmatrix} P^1 \\ P^2 \\ P^3 \\ P^4 \end{bmatrix}^{t} - \begin{bmatrix} P^1 \\ P^2 \\ P^3 \\ P^4 \end{bmatrix}^{t-1} \qquad (5.26)$$

Examining the second order matrix formulation of the acoustic wave equation solution we can see an important parallel aspect for each time step. All spatial grid points at time t_{i+1} are calculated independently and require only information from time t_i and t_{i-1} using a set of simultaneous equations.

Based on the observations above one can introduce parallel task allocation into each

grid point location. The ultimate goal would be to have as many processors as grid points, but the amount of calculation at each grid point is not floating point intensive enough to benefit from the capabilities of a MIMD computer. If one is using processors capable of doing a large number of floating point operations then one must assign tasks with an equivalent number of operations (at all x and y positions) at each time step.

This type of coarse parallelism was used to program the equation (5.13) in parallel. Parallelism runs over one of the spatial coordinates (z in our case), allowing the other spatial domains (x and y) to be calculated serially inside each task for higher efficiency, thus obtaining the time advanced 3-D grid points all simultaneously. This application is of great benefit to the solution of problems of this type. The faster a task is completed the smaller the time becomes between temporal steps.

Because of the need for information from more than one shot record for a typical survey, an outer level of parallelism will benefit the computations on an MIMD computer. I have included an extra parallel task over shots which has increased the floating point operations per node thus achieving higher performance. The table below describes the behaviour of the 3-D acoustic forward modeling algorithm on the Myrias SPS-2 and SPS-3 architectures.

TABLE 5.1 Performance monitoring 3-D acoustic modeling

SYSTEM	*Shots*	*Size*	*Time(sec.)*
SPARC330	*1*	*80x80x80*	*7920*
SPS-2 64	*1*	*80x80x80*	*990*
SPS-2 64	*15*	*80x80x80*	*8237*
SPS-3 32	*1*	*80x80x80*	*320*
SPS-3 32	*15*	*80x80x80*	*2630*

NOTE: All examples run over a total of 400 time steps.

5.6 Applications

The most obvious application of finite-difference modeling is to compute a shot record for a given geologic structure. However, for interpretation purposes, the more useful information is the stacked seismic section for such a structure, which can be obtained by computing a large number of shot records, each for a different shot location, followed by conventional data processing. For 2-D models, the seismic section can be computed at affordable costs. In the investigation of 3-D problems one needs to examine a number of seismic sections corresponding to different orientations of the 3-D structure. The following example shows the results of a 3-D model obtained by using the SPS hardware and the 3-D prestack migration algorithm described in Chapter 4.

Figure 5.2 shows the first interface below the surface of the model used for modeling. This is as step fault in 3-D and its two reflector levels are located at 450m and 500m below the surface. The receiver points were located at the surface grid k=0 and the source location was at x=41, y=41 and z=2. The following parameters were used for modeling. Grid spacing $\Delta x=\Delta y=\Delta z=10$m, grid dimensions 80x80x80, medium velocity above the fault 2400m/s and below at 3400m/s. Dominant frequency of the source signal 30Hz, time sampling of 1.5 ms with a total of 400 time steps. The shot data were stored at every 4 receiver positions thus generating a total of 20x20 (400) receivers per shot.

Figure 5.3 shows the complete shot record as calculated by the algorithm. The dominant features of this record are the direct wave arrival from the source to receivers and two dominant reflections from the model. A more detailed view (Figure 5.4) of a record at the y=41 (a line running across the middle of the model) shows two reflections originating at t=450ms and t=600ms. There is also a diffracting wave present due to the edge reflections from the fault. Another cross sectional view in Figure 5.5 for x=20,41 and 60 shows the reflecting signal from one of the model boundaries but also side reflection

interference from the side of the fault. Clearly, a complete three-dimensional diffraction pattern from the fault edge is present in the data (Figure 5.6) indicating the higher accuracy of the method as compared with 2-D modeling algorithms. Figure 5.7 shows three time slices of the wavefield. Each of these slices represents a snapshot of the wavefield reaching the horizontal plane z=41 (X-Y) and two vertical planes at x=41 (Y-Z) and y=41 (X-Z) at a time of 200ms. These time slices show the character of the wave as it propagates through the model. The reflection and transmission at the boundaries of the fault model are evident indicating that the algorithm simulates the seismic wave phenomena in 3-D appropriately.

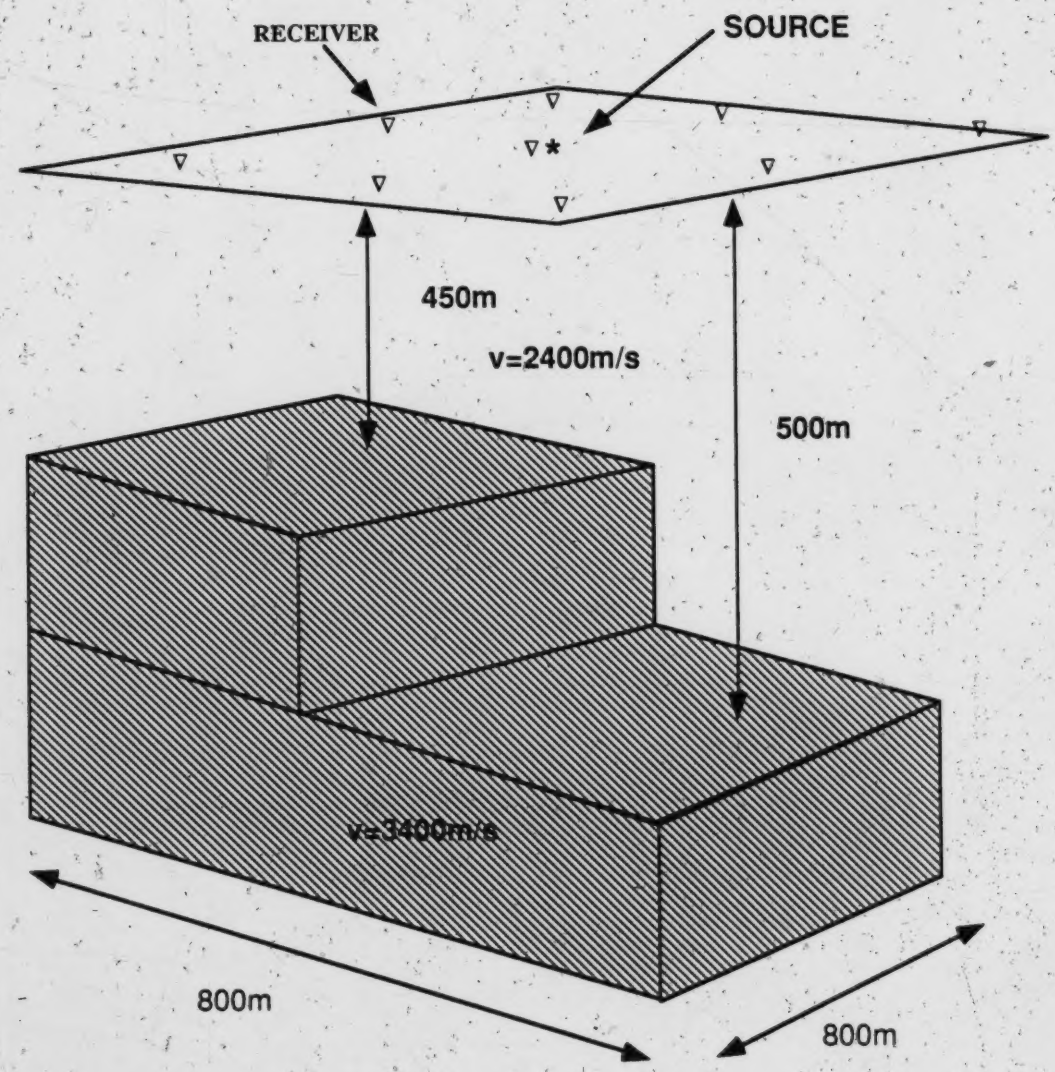


Figure 5.2 A perspective plot showing the 3-D step fault model used for modeling

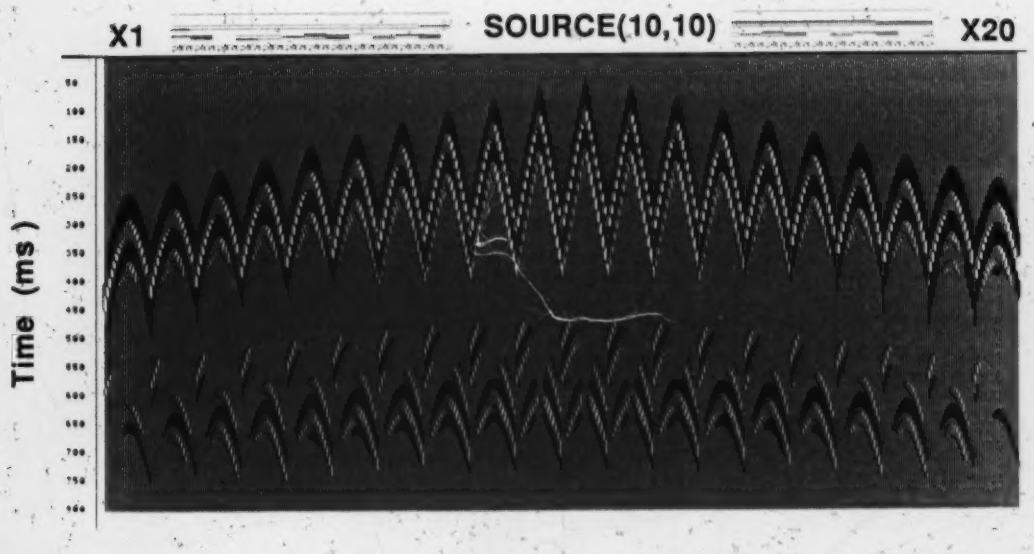


Figure 5.3 A synthetic 3-D shot record generated over the step fault model.

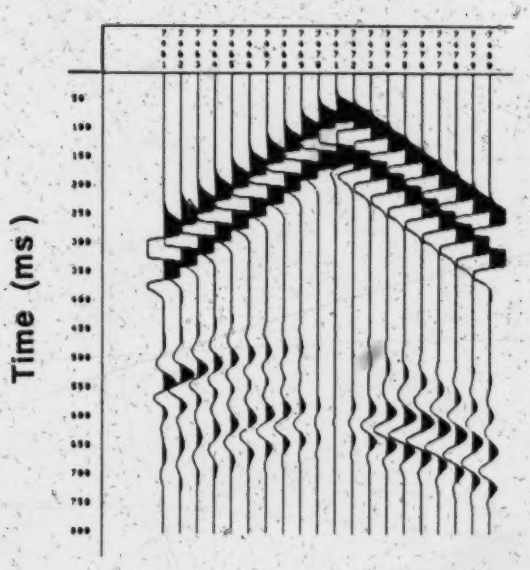


Figure 5.4 A vertical view of y=41 from the shot in Figure 5.3. The diffractions from the fault edge are evident.

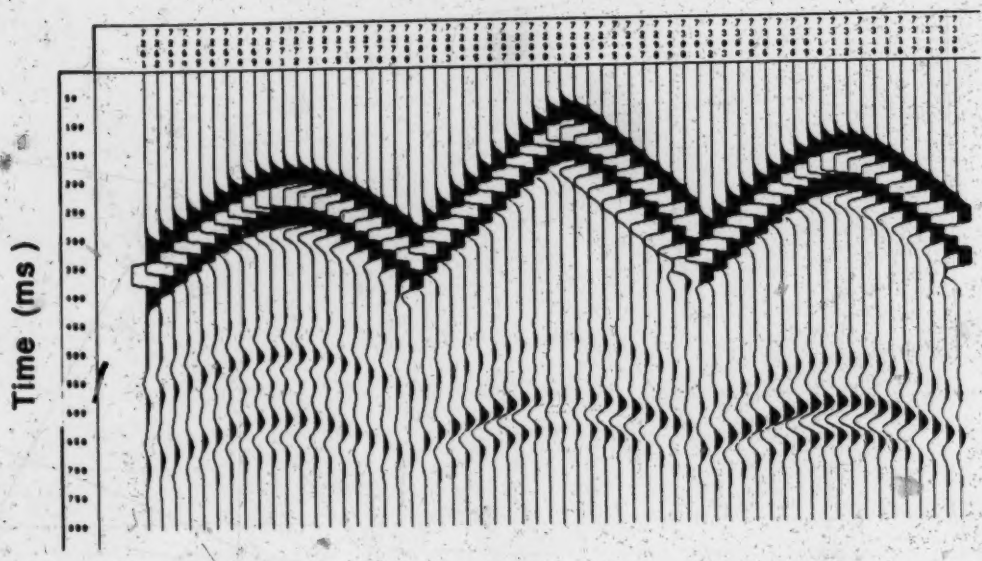
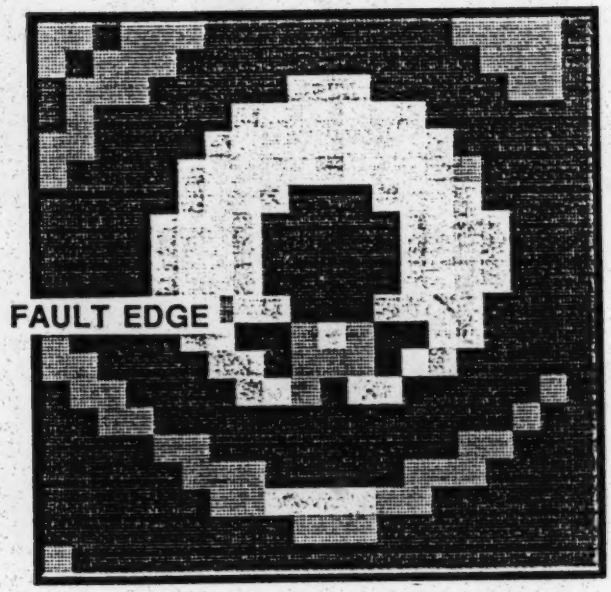


Figure 5.5 A vertical view of x=20, 41 and 60 from the shot in Figure 5.4. Diffractions from the sides of the fault are present.



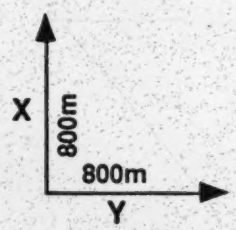


Figure 5.6 A horizontal view at t=500ms from the shot in Figure 5.4. The reflection character of the fault has been modeled properly.

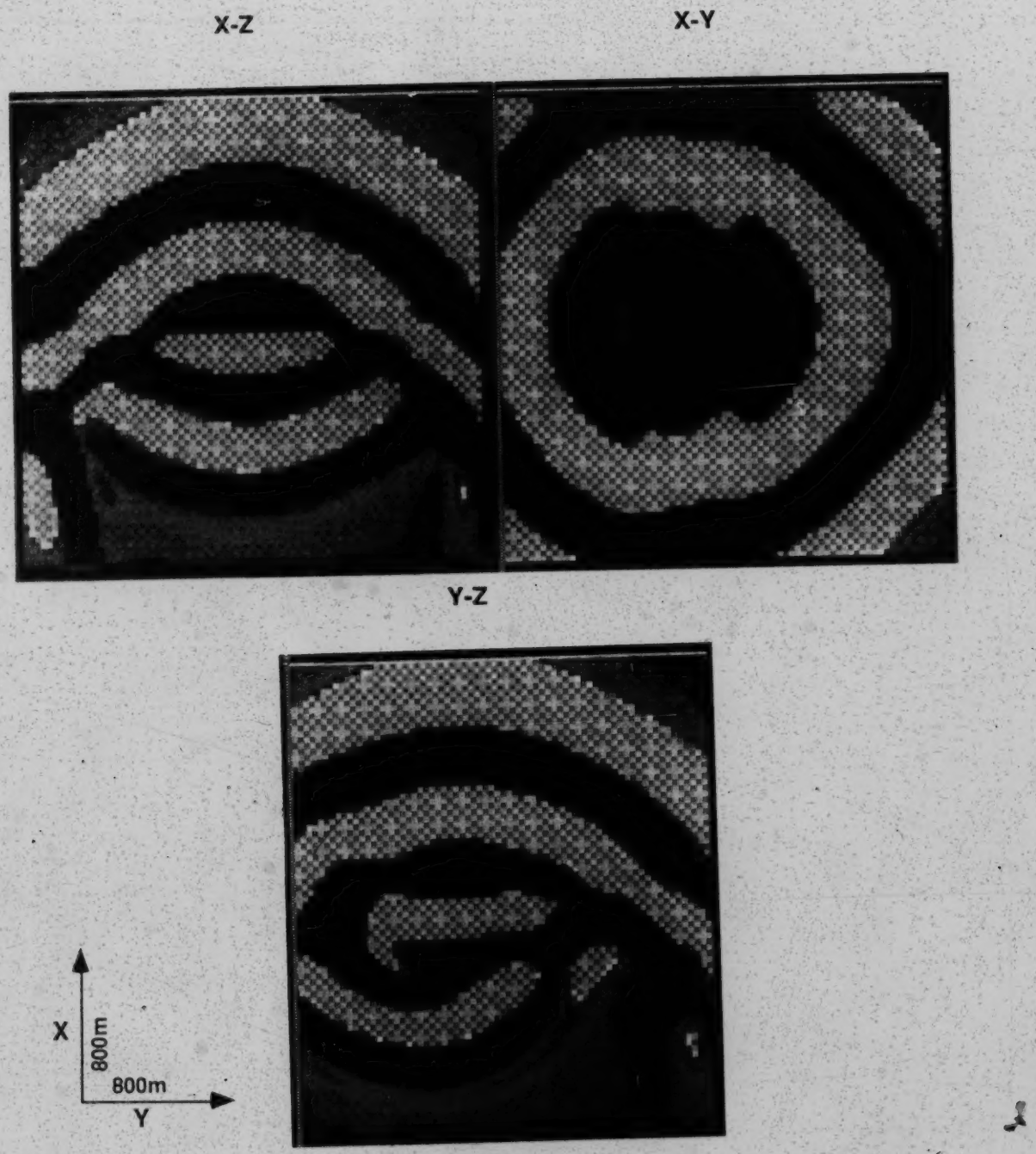


Figure 5.7 Snapshot for t=200ms. X-Y, X-Z and Y-Z planes showing the propagation character of the wavefield in 3-D media.

5.7 Some concluding remarks

Three-dimensional seismic modeling has been an impossible dream for geophysicists. The introduction of supercomputers and, recently, parallel ones has brought this dream closer to realization. This work was an attempt to investigate wave equation analysis of the seismic response of an arbitrary-shaped 3-D structure via parallel computing. The results indicate that much valuable information is lost if we treat the subsurface as a layered medium or a 2-D structure. The information from a 3-D seismogram is of utmost value to the seismic interpreter enabling him to examine the spatial configuration of the wavefield as it reaches the surface at different times.

CHAPTER 6

APPLICATION OF EDGE DETECTION TO SEISMIC IMAGES

6.1 Introduction

Edge-detection techniques have been widely applied in the field of image processing. Their application is almost unknown in seismic exploration, with no conventional applications to 2-D seismic data. However, moving from 2-D to 3-D seismic studies, planar images of time or depth slices of the wavefield may be characterized with edge detection methods to clarify the information of the subsurface sections. Edge detection could be the tool that reveals those characteristics that are not visible by other means.

In the words of Torre and Poggio (1986) "Edge detection is the process that attempts to characterize the intensity changes in the image in terms of the physical processes that have originated them". In their paper they describe the basic methods used for edge detection, which are primarily derivatives of different types, and possibly different scales, used to define those intensity changes by locating zero-crossings or maxima of a gradient. However, because of the *non regularity* of differentiation a regularizing filter operation is necessary. According to Torre and Poggio (1986) edge detection consists of two steps, namely a filtering step followed by a differentiation step. Canny (1986) describes a computational approach to edge detection constrained by the appropriate detector design based on the principle that edges are marked as maxima in gradient magnitude of a Gaussian function.

Huertas and Medioni (1986) used a method that detected edges on grey level images by finding zero-crossings in the convolution of the image with Laplacian-of-Gaussian (LoG) masks. This work shows the remarkable potential of the LoG method to locate edges with a precision depending on the signal-to-noise ratio. The implementation looks attractive

enough to be used for edge detection in seismic images.

Another strategy for edge detection is shown by Lacroix (1989) where she used an intermediate step in the method that generalizes the nonmaxima detected by the algorithm. Clark (1989) shows the limitations of the zero-crossing algorithms and proposed a method that would "authenticate" the correct ones. Nevertheless, there is a wide range of literature that involves pattern analysis of images in terms of edge detection which goes beyond the scope of this work.

I decided to use the Huertas and Medioni approach and develop an algorithm based on the application to the image by the Laplacian of Gaussian. The method was then applied to seismic images and produces some very interesting results. Those results are shown in Chapter 7. Here I will only give a description of the method and discuss the algorithm development.

6.2. Zero-Crossing of Laplacian-Gaussian Filtered Images.

A second-derivative edge-detection operator has been developed that uses a 3x3 pixel operator. That is, we first locate zero-crossings with pixel precision marking the edge of the pixel that has the smallest absolute value. Then, for the eight neighbors around this edge point, using a small window for each point, we fit a parametric polynomial function. We can then create a grid several times finer in the row and column directions, in which pixel values are the values of the analytic function sampled at this point. On this grid zero-crossings can be extracted with pixel precision.

We start by defining the Laplacian of the Gaussian (LoG) convolution mask as suggested by Marr (1980):

$$\nabla^2 G(x,y) = \frac{1}{2\pi\sigma^4}\left(2-\left(\frac{(x^2+y^2)}{\sigma^2}\right)\right).exp\left[-\frac{(x^2+y^2)}{2\sigma^2}\right] \qquad (6.1)$$

where σ is the space constant of the Gaussian, we then define as $w = 2\sqrt{2\sigma}$ the width of the central acitatory region of the operator. Usually the size of this operator is 3w, or 8.5σ; 99.7 percent of the area under a 1-D Gaussian lies between plus and minus 3 standard deviations from the mean, giving an area very close to zero of the $\nabla^2 G$. Figure 6.1 shows an illustration of these parameters.

Intensity changes in the image are detected in an image I(x,y) by finding the zero-crossings in $\nabla^2 G(x,y) * I(x,y)$, where * denotes convolution. Whenever an intensity change occurs, there is a peak in the first directional derivative of the intensity and a zero-crossing in the second directional derivative. Figure 6.2 shows a schematic representation of the intensity profile of an ideal step edge taken in the direction d_{max} for which the slope measured at the corresponding zero-crossing is maximum. Detection of intensity changes is then reduced to finding the zero-crossings in the second derivative by a polynomial function fitting.

Zero-crossings give a good localization of edges, assuming that in the neighborhood of a zero-crossing the filtered image can be well modeled by a polynomial. Subpixel values are then obtained by sampling this continuous function on a regular grid whose size depends on the desired resolution. Here, the interpolating function is a polynomial in the image row and column coordinates.

We assume that in a 3x3 neighborhood, the polynomial takes the form (Huertas,1986):

$$f(r,c) = k_0 + k_1 r + k_2 c + k_3 r^2 + k_4 rc + k_5 c^2 + k_6 r^2 c + k_7 c^2 r + k_8 r^2 c^2 \qquad (6.2)$$

where the computation of the coefficients is done using a Chebyshev discrete orthogonal polynomial set P_n.

Let I_{LoG} be the image convolved with the LoG filter. For each zero-crossing at location (r,c) in I_{LoG}, the fitting problem is to determine the coefficients $k_0,\ldots,k_8$ such that

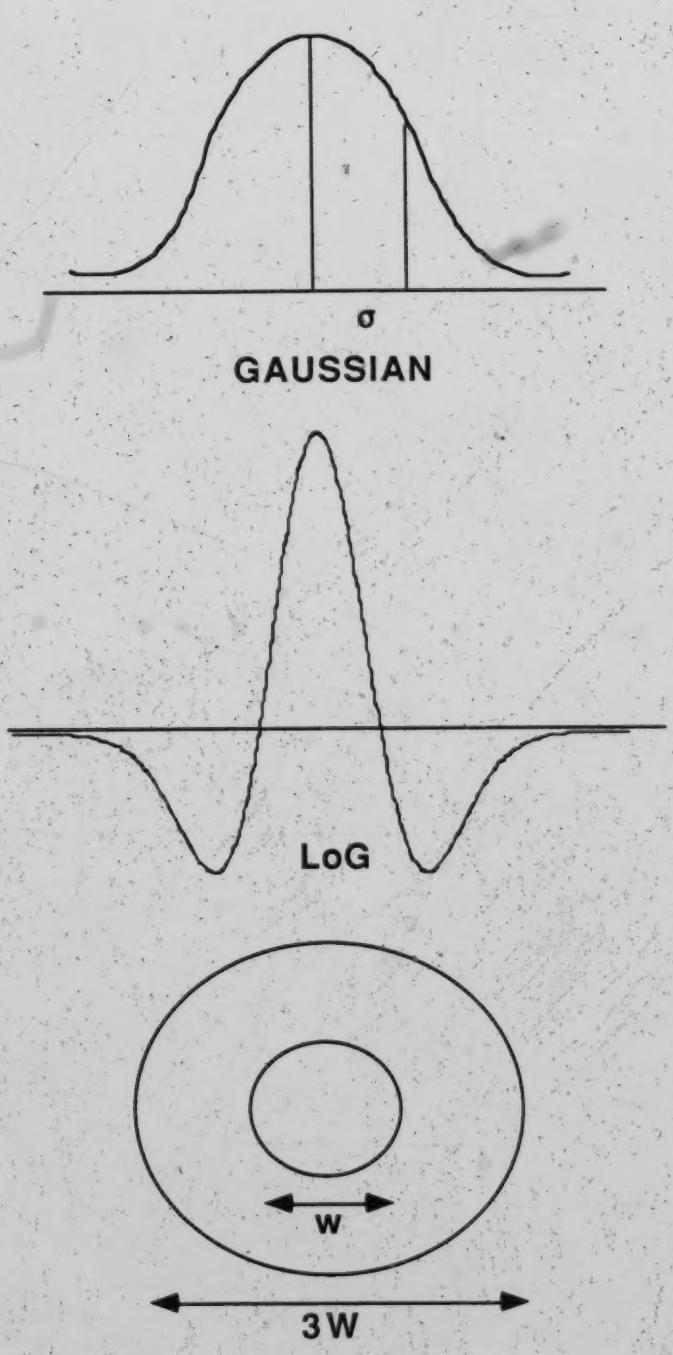


Figure 6.1 Parameters related to the LoG (Laplacian of Gaussian) operator.

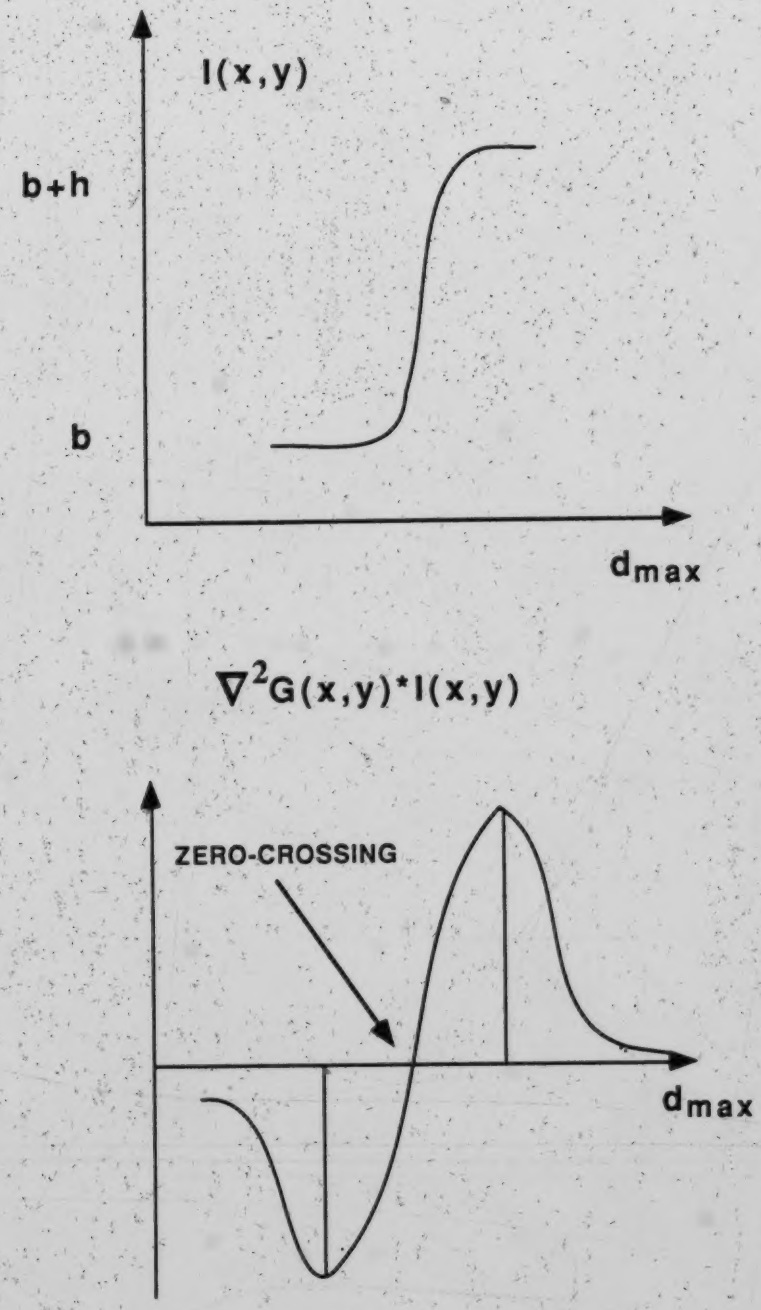


Figure 6.2 Response of a step to a LoG mask.

$$I_{LoG}(r,c) = \sum_{n=0}^{8} k_n P_n(r,c) \qquad (6.3)$$

If W represents a small window containing a zero-crossing, then with r,c,i,j, in W, these coefficients are given by:

$$k_n = \frac{\sum_r \sum_c P_n(r,c) I_{LoG}(r,c)}{\sum_i \sum_j P_n^2(i,j)} \qquad (6.4)$$

The masks for the computation of the coefficients vary. Here, I used a 3x3 window with the masks given by Huertas and Medioni (1986 pg. 656).

6.3 Algorithm implementation

To compute the zero crossings with subpixel accuracy in a seismic image I carried out the following steps during algorithm design.

1. *Convolve the image I of size R x C with LoG filter of size M^2, and obtain a filtered image of I_{LoG} of size R x C.*
2. *Compute a zero-crossings image I_{zxings} of size R x C by locating the zero-crossings in I_{LoG}, using the points where the response goes through a zero in a 3x3 window.*
3. *For each zero-crossing detected and marked as an edge in I_{zxings} and for its eight neighbors, compute the real valued function f_{LoG} defined by () over a real plane.*
4. *Compute a new zero-crossing image F_{zxings} of size nR x nC by locating the zero-crossings in a digital grid determined by the desired subpixel resolution n, over this plane. This part was left optional in the algorithm.*

These are the basic steps of the algorithm developed and tested on a 3-D seismic data set described in Chapter 7.

6.4 Conclusion

I have developed an algorithm based on principles of edge detection used in studies of pattern analysis and image processing. This was an initial investigation of the behaviour of such methods in relation to seismic image analysis. To my knowledge there has been no previous application of edge detection methods in seismology. The method seems to improve seismic image vision by delineating characteristics based on sudden amplitude variation in the sections. Direct applications of the method are shown in the next Chapter.

CHAPTER 7

THE 3-D DEPTH MIGRATION

7.1 DIRECT 3-D POSTSTACK DEPTH MIGRATION

7.1.1 Introduction - The seismologist's dream

Seismologists dream of the day when it will be possible to obtain automatically the Earth's physical properties directly from the recordings of seismic waves via exact 3-D processing. The importance of 3-D imaging has been discussed in previous sections but practical considerations regarding approximations to the solution of the 3-D problem introduce unwanted errors in the migrated results.

At the 1985 S.E.G. (Society of Exploration Geophysics) Convention in Washington, D.C. , A.J. Berkhout from Delft University, indicated that " even with the new generation of fast computers 3-D pre-stack migration is still unthinkable". It was generally accepted at the time that one must make compromises in order to achieve this goal. Since 1985 we have witnessed larger steps towards performance improvement in computing. New computers have paved the way for breaking up the computation into parallel components.

In general, 3-D pre-stack migration has been used under some dynamic assumptions as I have explained in Chapter 4. Those alternative methods of true 3-D are feasible within the limits of the present-day field recording parameters but as survey resolution increases more accurate processing schemes are needed. However, the exact solution of the 3-D migration problem is not mentioned by anyone in the literature to date and will require several Gigaflops of computer power to be feasible for exploration purposes.

Utilizing existing parallel computer technology and constraining the problem size to fit

the physical memory of the computing system, I decided to solve the exact 3-D migration problem at an initial phase thus examining its limitations and potentials.

This Chapter is devoted to the solution of the 45-degree 3-D migration equation using the direct solution for an enneadiagonal (9-diagonal) system of linear equations. The algorithm has been tested for stability and accuracy and applied to synthetic and real seismic data examples. The results have shown that if the appropriate computer power is available the method may be used for 3-D seismic imaging in areas of complex geological environments and when high-resolution results are needed.

7.1.2 The method

We start with the same scalar wave equation in 3-D as shown in Chapter 4:

$$\frac{\partial^2}{\partial x^2} P(x,y,z,t) + \frac{\partial^2}{\partial y^2} P(x,y,z,t) + \frac{\partial^2}{\partial z^2} P(x,y,z,t) = \frac{1}{v^2(x,y,z)} \frac{\partial^2}{\partial t^2} P(x,y,z,t) \quad (7.1)$$

The Fourier transformed field given by:

$$P(k_x,k_y,z,\omega) = \int \int \int P(x,y,z,t) e^{(ik_x x + ik_y y - i\omega t)} dx\, dy\, dt \quad (7.2)$$

leads to:

$$\frac{\partial^2 P}{\partial z^2} + \left\{ \left[\frac{\omega^2}{v^2(x,y,z)} - k_x^2 \right] + \left[\frac{\omega^2}{v^2(x,y,z)} - k_y^2 \right] \right\} P = 0 \quad (7.3)$$

It can easily be shown that the solution to this equation is also the solution to the one-way scalar wave equation:

$$\frac{\partial P}{\partial z} = -i\frac{\omega}{v}\left[\sqrt{1-\frac{v^2(x,y,z)\ k_x^2}{\omega^2}}+\sqrt{1-\frac{v^2(x,y,z)\ k_y^2}{\omega^2}}\right]P \qquad (7.4)$$

The vertical wavenumber is then given by:

$$k_z(z) = \pm\frac{\omega}{v}\left[\sqrt{1-\frac{v^2(x,y,z)\,k_x^2}{\omega^2}}+\sqrt{1-\frac{v^2(x,y,z)\,k_y^2}{\omega^2}}\right] \qquad (7.5)$$

As a next step we need to account for lateral velocity variations and examine the solution space in terms of this equation where the horizontal wavenumbers can be substituted by $(-ik_x 2) = \partial 2/\partial x 2$ and $(-ik_y 2) = \partial 2/\partial y 2$.

In order to bring equation (7.5) into the spatial domain an efficient and accurate evaluation of the square root of a differential operator is needed. I have approximated the square root using a suggested expression by Francis Muir (Clayton, 1980; Claerbout 1985) that is:

$$k_z = -\frac{\omega}{v}\left[\left(1-\frac{v(x,y,z)^2k_x^2}{\omega^2}\right)^{1/2}+\left(1-\frac{v(x,y,z)^2k_y^2}{\omega^2}\right)^{1/2}-1\right] \qquad (7.6)$$

Which leads to the 45 degree approximation of the 3-D wave equation. Substituting the continued fraction expansion of equation (7.6) into equation (7.4) we get:

$$\frac{\partial P}{\partial z} = -i\left(\frac{\omega}{v(z)}-\frac{k_x^2}{\frac{2\omega}{v(x,y,z)}-\frac{v(x,y,z)k_x^2}{2\omega}}-\frac{k_y^2}{\frac{2\omega}{v(x,y,z)}-\frac{v(x,y,z)k_y^2}{2\omega}}\right)P \qquad (7.7)$$

invoking the retarded coordinate system principle (Chapter 3) the wavefield *P* becomes:

$$\frac{\partial P(x,y,z,\omega)}{\partial z} = \left(\frac{\partial}{\partial z} - \frac{i\omega}{v(z)}\right) Q(x,y,z,\omega)\, e^{\,i\omega \int_0^z \frac{dz}{v(z)}} \qquad (7.8)$$

putting equation (7.8) into equation (7.7) we end up with a differential equation of the form,

$$\frac{\partial Q(x,y,z,\omega)}{\partial z} = i\frac{k_x^2}{\frac{2\omega}{v(x,z)} - \frac{v(x,z)\,k_x^2}{2\,\omega}} Q(x,y,z,\omega) + i\frac{k_y^2}{\frac{2\omega}{v(x,z)} - \frac{v(x,z)\,k_y^2}{2\,\omega}} Q(x,y,z,\omega)$$

$$- i\,\omega\left(\frac{1}{v(x,y,z,)} - \frac{1}{v(z)}\right) Q(x,y,z,\omega) \qquad (7.9)$$

The first two terms on the right hand side represent the *diffraction term*. The third term on the right is called the *thin lens* term. In the derivation above we assume that $v(x,y,z)$ is a locally constant medium velocity and $v(z)$ depends on depth below the source, only.

Equation (7.9) may be rearranged by substituting for:

$-k_x^2 = \partial^2/\partial x^2$, $-k_y^2 = \partial^2/\partial y^2$, and $\mu(x,y,z) = \omega/v(x,y,z)$,

The diffraction portion of the solution is given by the following approximation:

$$-2\,i\,\mu(x,y,z)\frac{\partial Q(x,y,z,\omega)}{\partial z} - \frac{i}{2\,\mu(x,y,z)}\left[\frac{\partial^3}{\partial x^2\,\partial z} + \frac{\partial^3}{\partial y^2\,\partial z}\right] Q(x,y,z,\omega)$$

$$+\left[\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right] Q(x,z,\omega) + \frac{1}{2\mu^2(x,y,z)}\frac{\partial^4}{\partial x^2 \partial y^2} Q(x,y,z,\omega) = 0 \qquad (7.10)$$

This equation will be evaluated step by step , starting at the 2-D surface and moving to any depth, using a finite difference approximation. The fifth order derivative term has been dropped to avoid the extra computational effort needed to solve (7.10). This approximation will be described in detail in the next section.

7.1.3 Implicit finite - difference solution of the diffraction term

Because of poor stability, explicit difference methods are rarely used to solve initial boundary value problems in two or more space dimensions. Implicit methods with their superior stability properties are almost always used. Unfortunately, an implicit method in two space dimensions requires a set of equations to be solved at the advanced depth level simultaneously. First, I will show the development of an implicit scheme that solves (7.10) and second, in Section 7.1.4 describe the direct solution of the derived system of equations.

Rewrite equation (7.10) as

$$-2i\mu(x,y,z)\,\delta_z Q(x,y,z,\omega) - \frac{i}{2\mu(x,y,z)}\left[\delta_{xx}\delta_z + \delta_{yy}\delta_z\right]Q(x,y,z,\omega)$$
$$+\left[\delta_{xx} + \delta_{yy}\right]Q(x,z,\omega) + \frac{1}{2\mu(x,y,z)^2}\delta_{xx}\delta_{yy}Q(x,y,z,\omega) = 0 \qquad (7.11)$$

Substitute (see Appendix C) :

$$\delta_{xx} = \frac{1}{\Delta x^2}\left(\frac{\delta^2}{1+\gamma\delta^2}\right) \text{ and } \delta_{yy} = \frac{1}{\Delta y^2}\left(\frac{\delta^2}{1+\gamma\delta^2}\right) \qquad (7.12)$$

where Δx, Δy is the grid spacing and $\gamma = 1/12$ (Mitchell and Griffiths, 1980), where the δ^2 and δ^4 operators that involve the cross terms of x,y are defined respectively as:

$$\begin{bmatrix} & 1 & \\ 1 & -4 & 1 \\ & 1 & \end{bmatrix} \qquad \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

An implicit Crank-Nicolson (Mitchell and Griffiths, 1980) finite difference algorithm scheme is employed to solve the difference equation. This algorithm is unconditionally stable (Mitchell and Griffiths, 1980; Appendix E) and involves more than one point at the advanced depth level according to :

$$q^n_{m,l} = \frac{1}{2}\left(q^{n+1}_{m,l} + q^n_{m,l} \right) \tag{7.13}$$

The finite difference approximation of the differential equation (7.11) employing equations (7.12) and (7.13) reads:

$$\begin{aligned} &-\frac{2\mu i}{\Delta z}\left(q^{m+1}_{n,l} - q^m_{n,l}\right) - \frac{i}{2\Delta x^2 \mu}\left(\frac{\delta^2}{1+\gamma\delta^2}\right)\left(\frac{q^{m+1}_{n,l} - q^m_{n,l}}{\Delta z}\right) \\ &-\frac{i}{2\Delta y^2 \mu}\left(\frac{\delta^2}{1+\gamma\delta^2}\right)\left(\frac{q^{m+1}_{n,l} - q^m_{n,l}}{\Delta z}\right) + \frac{1}{2}\frac{\delta^2}{\Delta x^2(1+\gamma\delta^2)}\left(q^{m+1}_{n,l} + q^m_{n,l}\right) \\ &+\frac{1}{2}\frac{\delta^2}{\Delta y^2(1+\gamma\delta^2)}\left(q^{m+1}_{n,l} + q^m_{n,l}\right) + \frac{1}{4\mu^2\Delta x^2\Delta y^2}\left[\frac{\delta^4}{\left(1+\gamma\delta^2\right)^2}\right]\left(q^{m+1}_{n,l} + q^m_{n,l}\right) = 0 \end{aligned} \tag{7.14}$$

Assuming that $\Delta x = \Delta y$ and multiply through by $\Delta x^2\left(1+\gamma\delta^2\right)^2$ to get:

$$-\frac{2\mu i\Delta x^2}{\Delta z}(1+\gamma\delta^2)(q_{n,l}^{m+1}-q_{n,l}^{m})-\frac{i}{\Delta z\,\mu}\delta^2(1+\gamma\delta^2)\left(\frac{q_{n,l}^{m+1}-q_{n,l}^{m}}{\Delta z}\right)$$
$$+\ \delta^2(1+\gamma\delta^2)\left(q_{n,l}^{m+1}+q_{n,l}^{m}\right)+\frac{1}{4\mu^2\Delta x^2}\delta^4\left(q_{n,l}^{m+1}+q_{n,l}^{m}\right)=0 \qquad (7.15)$$

Define the coefficients as,

$$A=\frac{i}{\mu\,\Delta z},\quad B=\frac{2\mu i\Delta x^2}{\Delta z}\ \text{and}\ C=\frac{1}{4\mu^2\Delta x^2} \qquad (7.16)$$

and separate the terms for m+1 and m in (7.15) and merge the common δ terms to get:

$$-Bq_{n,l}^{m+1}+\left(-2B\gamma-A+1\right)\delta^2 q_{n,l}^{m+1}+\left(-\gamma^2B-\gamma A+\gamma+C\right)\delta^4 q_{n,l}^{m+1}=$$
$$-Bq_{n,l}^{m}+\left(-2B\gamma-A-1\right)\delta^2 q_{n,l}^{m}+\left(-\gamma^2B-\gamma A-\gamma-C\right)\delta^4 q_{n,l}^{m} \qquad (7.17)$$

apply the operators δ^2 and δ^4 and write in expanded form as left hand side (LHS) and right hand side (RHS)

$$LHS=$$
$$-Bq_{n,l}^{m+1}+\left(-2B\gamma-A+1\right)\left(q_{n-1,l}^{m+1}+q_{n+1,l}^{m+1}-4q_{n,l}^{m+1}+q_{n,l-1}^{m+1}+q_{n,l+1}^{m+1}\right)+\left(-\gamma^2B-\gamma A+\gamma+C\right)$$
$$\left(q_{n-1,l-1}^{m+1}+q_{n-1,l+1}^{m+1}-2q_{n-1,l}^{m+1}-2q_{n+1,l}^{m+1}+4q_{n,l}^{m+1}-2q_{n,l-1}^{m+1}-2q_{n,l+1}^{m+1}+q_{n+1,l-1}^{m+1}+q_{n+1,l+1}^{m+1}\right) \qquad (7.18)$$

and after merging of common terms:

$$LHS=q_{n,l}^{m+1}\left(-B+8\gamma B+4A-4-4\gamma^2B-4\gamma A+4\gamma+4C\right)+$$
$$\left(q_{n-1,l}^{m+1}+q_{n+1,l}^{m+1}+q_{n,l-1}^{m+1}+q_{n,l+1}^{m+1}\right)\left(-2\gamma B-A+1+2\gamma^2B+2\gamma A-2\gamma-2C\right)$$
$$+\left(q_{n-1,l-1}^{m+1}+q_{n-1,l+1}^{m+1}+q_{n+1,l-1}^{m+1}+q_{n+1,l+1}^{m+1}\right)\left(-\gamma^2B-\gamma A+\gamma+C\right)$$

Similarly the RHS is given by:

$$RHS = q^m_{n,l}\left(-B+8\gamma B+4A+4-4\gamma^2 B-4\gamma A-4\gamma-4C\right)$$
$$+\left(q^m_{n-1,l}+q^m_{n+1,l}+q^m_{n,l-1}+q^m_{n,l+1}\right)\left(-2\gamma B-A-1+2\gamma^2 B+2\gamma A+2\gamma+2C\right)$$
$$+\left(q^m_{n-1,l-1}+q^m_{n-1,l+1}+q^m_{n+1,l-1}+q^m_{n+1,l+1}\right)\left(-\gamma^2 B-\gamma A-\gamma-C\right) \tag{7.19}$$

Replacing the coefficients of q as follows:

$$\overline{A}=\left(-2\gamma B-A-1+2\gamma^2 B+2\gamma A-2\gamma-2C\right)$$
$$\overline{B}=\left(-B+8\gamma B+4A-4-4\gamma^2 B-4\gamma A+4\gamma+4C\right)$$
$$\overline{C}=\left(-\gamma^2 B-\gamma A+\gamma+C\right)$$

and for the right hand side (7.20)

$$\overline{E}=\left(-2\gamma B-A-1+2\gamma^2 B+2\gamma A+2\gamma+2C\right)$$
$$\overline{D}=\left(-B+8\gamma B+4A+4-4\gamma^2 B-4\gamma A-4\gamma-4C\right)$$
$$\overline{F}=\left(-\gamma^2 B-\gamma A-\gamma-C\right)$$

Using the above defined coefficients we can rewrite equation (7.15) as a linear system of equations of the form:

$$\overline{B}\,q^{m+1}_{n,l}+\overline{A}\left(q^{m+1}_{n-1,l}+q^{m+1}_{n+1,l}+q^{m+1}_{n,l-1}+q^{m+1}_{n,l+1}\right)$$
$$+\overline{C}\left(q^{m+1}_{n-1,l-1}+q^{m+1}_{n-1,l+1}+q^{m+1}_{n+1,l-1}+q^{m+1}_{n+1,l+1}\right)=$$
$$\overline{D}\,q^{m1}_{n,l}+\overline{E}\left(q^{m1}_{n-1,l}+q^{m}_{n+1,l}+q^{m}_{n,l-1}+q^{m}_{n,l+1}\right)$$
$$+\overline{F}\left(q^{m}_{n-1,l-1}+q^{m}_{n-1,l+1}+q^{m}_{n+1,l-1}+q^{m}_{n+1,l+1}\right) \tag{7.21}$$

In the above equation the $m+1$ terms are unknown and the m terms present known values. The system of equation to be solved is of the enneadiagonal form and can be

solved directly using complex sparse-matrix solver techniques. These sets of equations can be solved using the principle of under-relaxation discussed in Chapter 4 to obtain a fast solution but it has been left for future research and goes beyond the scope of this research.

To solve an enneadiagonal system of equations directly is normally prohibitively expensive, but with the aid of a parallel computer and using fairly small data sets, solutions can be obtained in a reasonable time frame. The computational time needed to solve this system will be discussed later in this Chapter for each application involved. Meanwhile, I shall briefly discuss the direct method used to solve this system using a math library routine (CSPSLV).

7.1.4 Complex sparse-matrix solution and parallel implementation

There are numerous methods in the literature that can be used for the solution of complex sparse matrices. These could include complex matrix inversion which becomes unrealizable for large system of equations, conjugate gradient or conjugate direction methods that seem attractive (I will be considering their application as an extension of the present work) and Gaussian elimination based solvers that can be used for moderate size matrices. The goal of this work was not to try to determine the most efficient of the methods but rather obtain a solution of the enneadiagonal sparse matrix with a library routine and leave the investigation for a faster solver open to future research.

I have decided to use a public library routine called CSPSLV to solve equation (7.21) which uses sparse Gaussian elimination with column interchanges to solve a complex linear system **Ax=B**. The elimination phase performs row operations on **A** and **B** to obtain a unit upper triangular matrix **U** and a vector **Y**. The solution phase solves **Ux=Y**

The method requires minimal storage because the sparse matrix **A** is stored using three arrays, two pointers and a data array. The data array contains the nonzeros of A stored

row-by-row, not necessarily in order of increasing column number. The pointer arrays contain information about the column and row number of the nonzero elements.

The algorithm is solved serially and no parallel version of it is available or has been developed by the author. Discussions on sparse matrix solvers for real positive definite matrices for the SPS-2 have been given by Joe (1990) and Stone et al. (1990). In any case, the solution of the system of equations for different frequencies allows us to solve serially many of these (limited by the number of processors) at the same time, thus gaining speed in the long run. Because the algorithm spends most of its time in the solver I believe that an improved parallel or serial solver for the direct solution will be beneficial to the feasible solution of the exact 3-D migration problem.

Performance evaluation of the method as compared with the solution via under-relaxation has shown a delay time of an order of magnitude. In other words, using the same hardware platform it takes 10 times longer to solve the exact 3-D problem.

The method has been solved in parallel over frequencies with the option of parallelism over the horizontal directions. Figure 7.1 shows a layout of the exact 3-D pseudo-code. The code ran very efficiently at 95% user time in both SPS-2 and SPS-3 hardware. Some estimate of run-times are shown in the following table.

TABLE 7.1 Performance monitoring exact 3-D ω-x-y migration

# PE's	Problem Size	SPS-2	SPS-3
32	30x30	6768 sec.	960 sec.
32	40x40	17,462 sec.	2800 sec.
32	45x45	26,525 sec.	3780 sec.

NOTE: The depth extrapolation for the above examples was 100 steps with an 8 m. sampling interval.

ω-X-Y TRUE 3-D MIGRATION

(MAIN PART OF PARALLELISM)

(PARALLEL) DO over Frequency steps

DO over Depth steps

(PARALLEL) DO over Y
DO over X

A. Calculate the time retarded field

$$Q(x,y,z,\omega) = P(x,y,z,\omega)\ e^{\,i\,\omega \int_0^z \frac{dz}{v(z)}}$$

B. Solve the diffraction term (9-diagonal)

$$-2\,i\,\mu(x,y,z)\frac{\partial Q(x,y,z,\omega)}{\partial z} - \frac{i}{2\,\mu(x,y,z)}\left[\frac{\partial^3}{\partial x^2\,\partial z} + \frac{\partial^3}{\partial y^2\,\partial z}\right] Q(x,y,z,\omega) + \left[\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right] Q(x,z,\omega)$$
$$+ \frac{1}{2\mu^2(x,y,z)}\frac{\partial^4}{\partial x^2 \partial y^2} Q(x,y,z,\omega) = 0$$

C. thin lens term

$$\frac{\partial Q(x,y,z,\omega)}{\partial z} = -\,i\,\omega\left(\frac{1}{v(x,y,z)} - \frac{1}{v(z)}\right) Q(x,y,z,\omega)$$

END DO over X
END DO over Y

END DO LOOP over depth
Imaging condition: $M(x,y,z) = \sum_{\omega} Q(x,y,z,\omega)$

END PARDO over frequencies

Figure 7.1 Exact 3-D 45 degree pseudocode.

7.1.5 The 3-D 45 degree response.

In order to evaluate the 3-D 45-degree solution of the wave equation I needed to analyze its response in steps similar to that followed in the previous chapters. The ideal 3-D response as we have seen in Chapter 4 is a hollow hemisphere. The 15 degree solution appeared to have some circular symmetry but the traveltimes were not as accurate. In this section, I used the same model as in Chapter 4, a point diffractor at 300 ms.

The impulse response of the exact 45 degree 3-D solution will be compared with the one obtained using the one-pass method and splitting of the migration operator. This one-pass code has been developed by the author but not mentioned in any other parts of the thesis. Figure 7.2 shows a cross-sectional view of (a) the desired response, (b) the exact 3-D response and (c) the one-pass 3-D response for y=20. Evidently, this approximation produces more accurate travel times than the 15 degree case. However, one should examine the planar view of the response to check for error in the azimuthal direction where the onepass method breaks down.

Figure 7.3 displays a 2-D slice of both the (a) exact and (b) the onepass methods. As expected, the one pass case shows anisotropic error in the diagonal directions. This is due to operator splitting and will probably introduce errors during migration. The exact 3-D case shows perfect circular behaviour in all directions with anisotropic effects.

The direct 3-D 45 degree solution of the wave equation has shown a very accurate response and justifiably will generate better migration results as compared with methods which use operator splitting or separation to reach an approximate solution. Although it is computer-intensive and time-consuming, the solution developed here generates more accurate migrated results in relation to the operator-splitting methods.

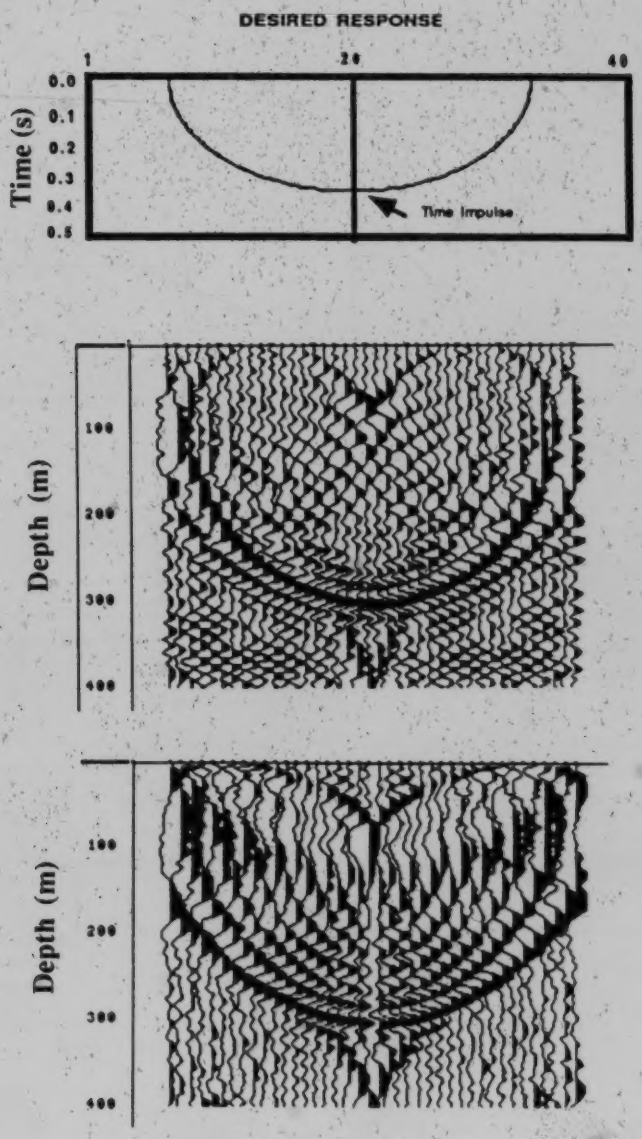


Figure 7.2 (a) Desired impulse response for migration, (b) 45^0 3D direct, (c) 45^0 3D onepass

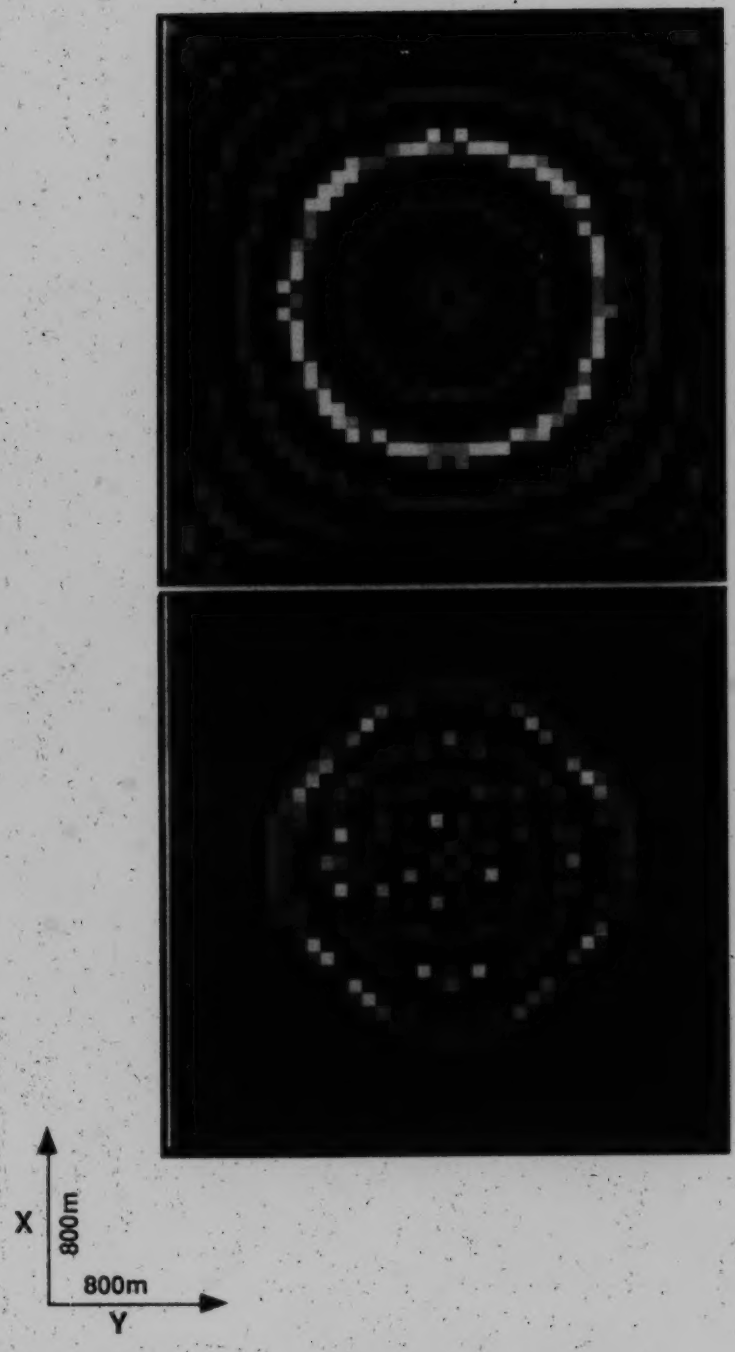


Figure 7.3 Depth slices at 200m for the response of (a) 45^0 3D direct, (b) 45^0 3D onepass.

7.1.6 3-D zero-offset Modeling and Migration

The concept of exploding reflectors (ERM) can be used for both 3-D zero-offset modeling and migration. The approach described in Chapter 4, Section 4.1.9.2, is used here to generate 3-D synthetic zero-offset responses in order to test the exact 45 degree solution versus the under-relaxed one. I have modified the migration algorithm developed in Chapter 4 to use the ERM principle in order to obtain true 3-D zero offset seismic sections over any physical model configuration. In principle the idea is based upon the upward (downward for migration) continuation of the wavefield, generated at the "exploding" reflectors, to the surface.

The model shown in Figure 4.8 was used to generate a 30 x 30 x 300 seismic volume that includes reflections from two dipping events for lines 1 to 9 and 21 to 30 and a syncline along with the same dipping reflectors between lines 10 and 20. Figure 7.4 shows three different cross-sections of this cube for lines 9,10,11,14,15 and 16. As can be seen the response does not resemble the physical characteristics of the model. 3-D migration is needed to image these events properly.

Figure 7.5 displays the 3-D migrated result for lines 8,9,10,14,15 and 16 respectively. It is clear that the method images the physical model correctly not only vertically but laterally as well. The extent of the syncline from line 10 to 20 has been preserved indicating the spatial accuracy of the method, contrary to the onepass case (Figure 4.10) where the syncline image extends beyond lines 10 and 20.

A 2-D planar view of the edges of the syncline is shown in Figure 7.6. The direct 3-D migration (a) has imaged the edges correctly, the onepass 3-D migration (c) indicates quite a spatial discrepancy in the image, a better approximation of it is shown in (b) obtained via the onepass under-relaxed case.

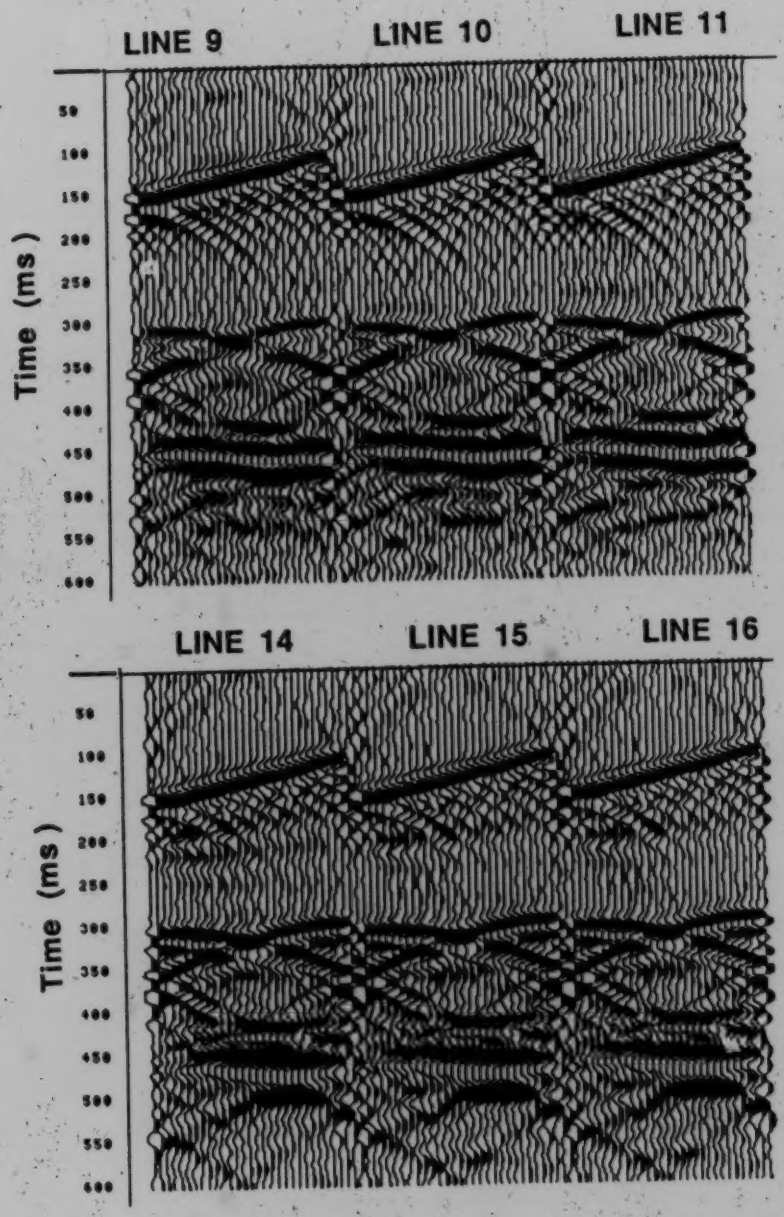


Figure 7.4 3D zero offset response for lines 9,10,11,14,15 and 16.

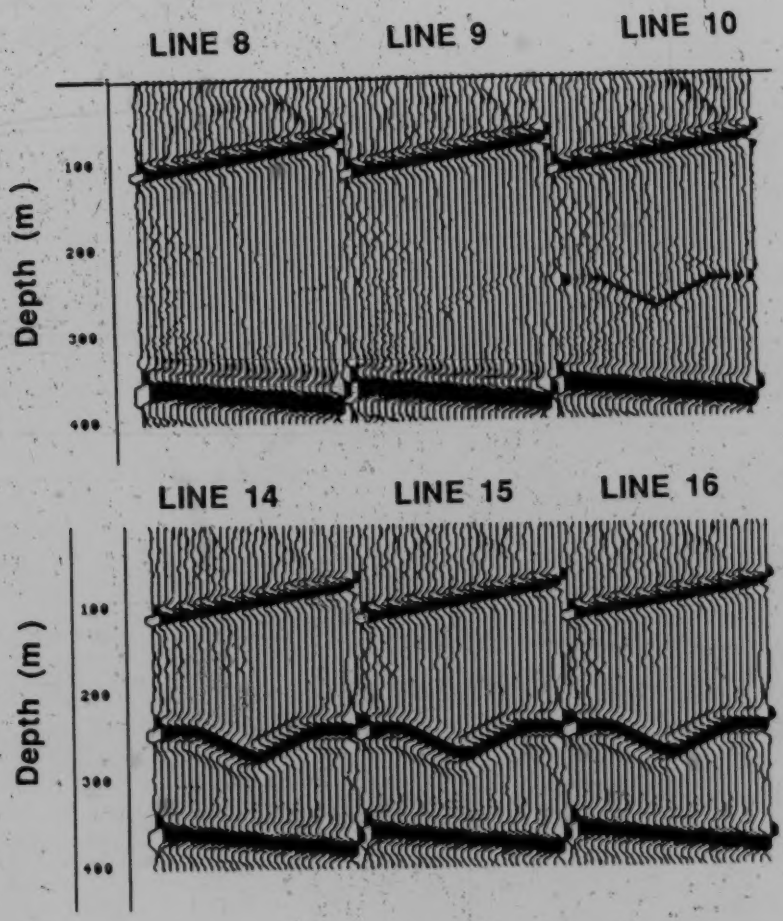


Figure 7.5 3-D migrated results for lines 8,9,10,14,15 and 16 respectively.

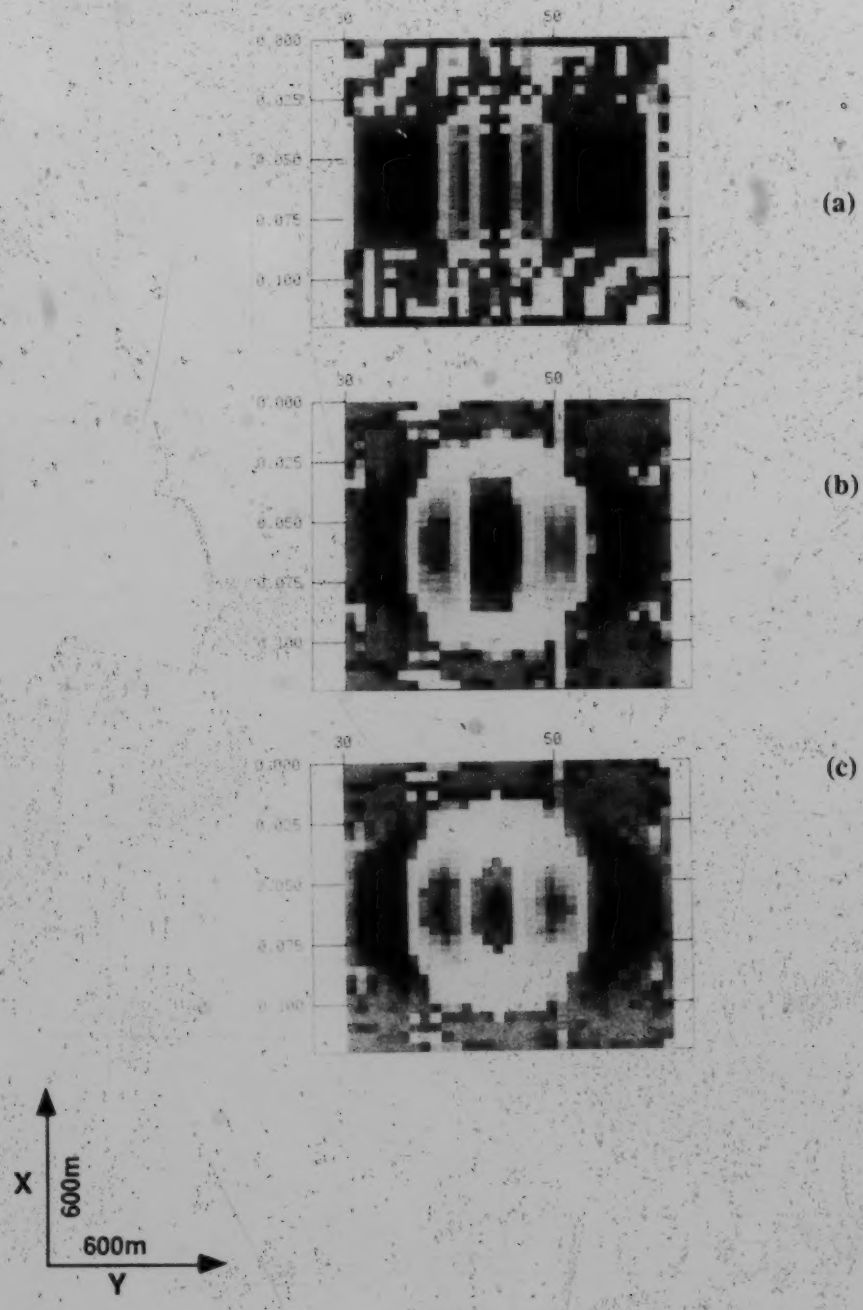


Figure 7.6 2D planar view of the edges of the syncline. Direct 3-D migration (a) onepass 3-D migration under-relaxed α=0.4, θ=0.65 (b) α=0.4, θ=0.45 (c).

7.2 3-D MIGRATION OF A SEISMIC DATA SET OVER A STEAM INJECTION REGION

7.2.1 Introduction

During 1985 and 1986, 3-D reflection data were acquired by Esso Resources Canada Limited over a heavy oil recovery zone near Cold Lake Alberta before and after steam injection and bitumen production. The purpose of the experiment was to investigate the feasibility of mapping steam affected reservoir zones using 3-D seismic methods. Esso provided this data to University of Alberta for analysis and interpretation

The experiment consisted of a 68 by 72 gridded stack lines (12.5m spacing) over a region of several injection/production wells. Initial recording was carried out before any steam injection activity and the second one during first cycle production (of cyclic steam stimulation) for most of the 22 wells. Four wells on the north row were undergoing second cycle steam injection. The data had been prestack processed by Esso and given to University of Alberta in poststack form. Previous analysis and interpretation by the company did not lead to clear evidence for sonic anomalies related to steam injection in the data. Here, I examine the data using methods developed during this research work.

The 3-D post stack migration with the under-relaxation and the exact 3-D migration methods were used to analyze the data in an attempt to delineate the presence of sonic anomalies related to steam injection in the zone of interest as well as understand the limitations of the approximate methods with respect to direct ones. These new migration methods have shown better resolution than other methods and have given an indication of some variation in the depth slices before and after steam injection. In order to improve the imaged sections and obtain a better definition of the anomalies, an image enhancement method along with an edge detection approach has been applied.

7.2.2 3-D processing and analysis

The location of the seismic grid and the wells are shown in Figure 7.7 on a plan view. There are a total of 22 steam injection/production wells covering an area of about 0.15 km^2 with the total survey spanning about 0.8 km^2. The final stacked section consisted of about 4896 traces in all. It was generated via 3-D stacking of 370 shots layout in a 10 x 37 grid with x direction spacing of 100m. and y direction spacing of 25m. The wavefield was recorded at variable receiver numbers laid out in a grid ranging from 92 to 332 with x direction spacing of 25m and y direction spacing of 100m.

The data were recorded for 2 s with a sampling interval of 1 ms. Figure 7.8 illustrates three of these stack sections for inlines (y direction) 44, 45 and 46. Plots on the left side of the Figure are from before steam and on the right are after steam. The region of interest is around 500ms corresponding to steam injection horizons. I chose to display these lines because they are close to the north row of 4 wells which were undergoing second cycle steam injection during the after steam seismic survey. Hence, there was a better chance of locating sonic anomalies for these lines.

Looking at the plots in Figure 7.8 one can see no major differences before and after the steam injection in the region of interset (~500ms). In particular, it is clear the the reflector above the injection zone (400ms) has not changed after steaming, whereas the reflector below the steam zone (600ms, Devonian formation) shows some discrepancies when compared before and after steam injection.

To study the above changes, I decided to migrate the data using both algorithms developed in this thesis. A cross-section of the velocity model used for this migration is shown in FIgure 7.9. Figure 7.10 shows the result of the exact 3-D migration method for the inlines 44,45 and 46 displayed before steam (a) and after steam (b). The cross-sections indicate some evidence of variation around the mean injection depth of wells (485m to

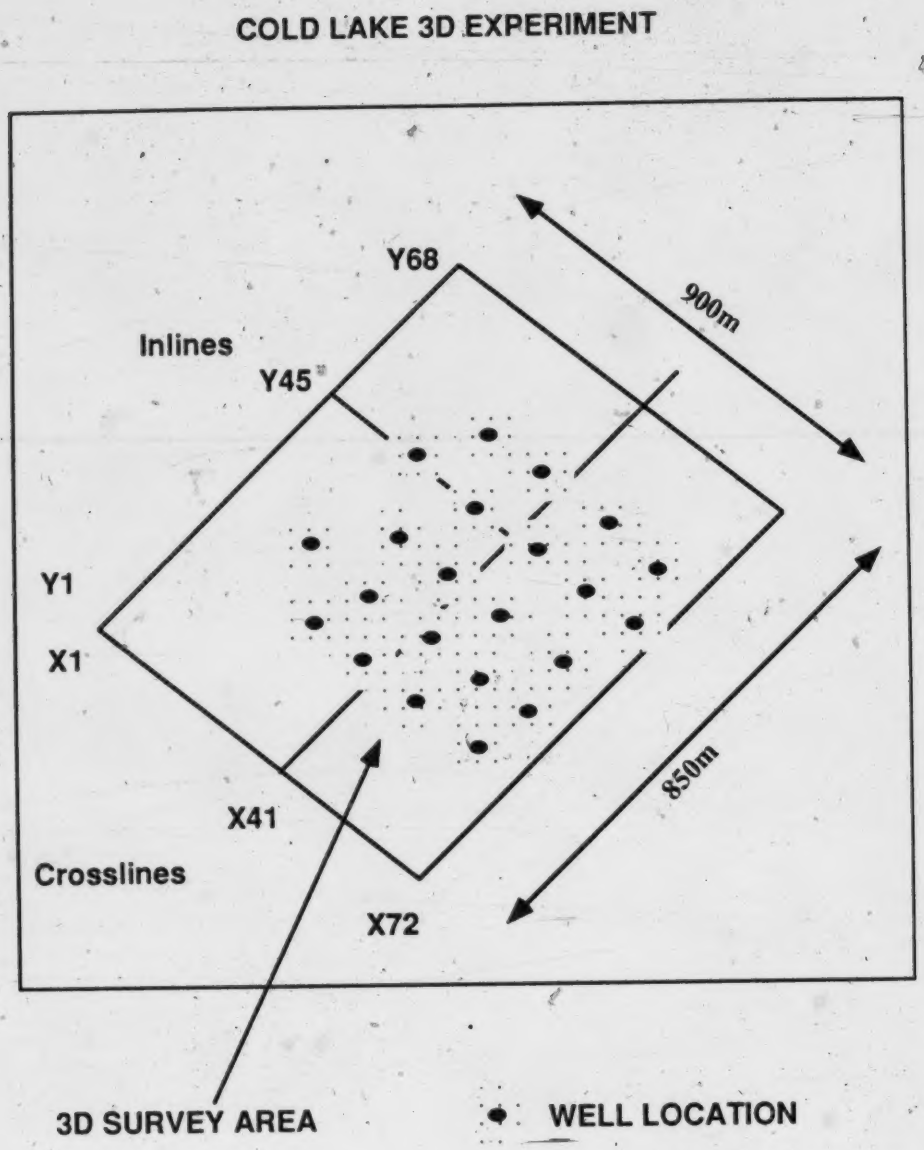


Figure 7.7 Plan view of the 3-D seismic survey in Cold Lake.

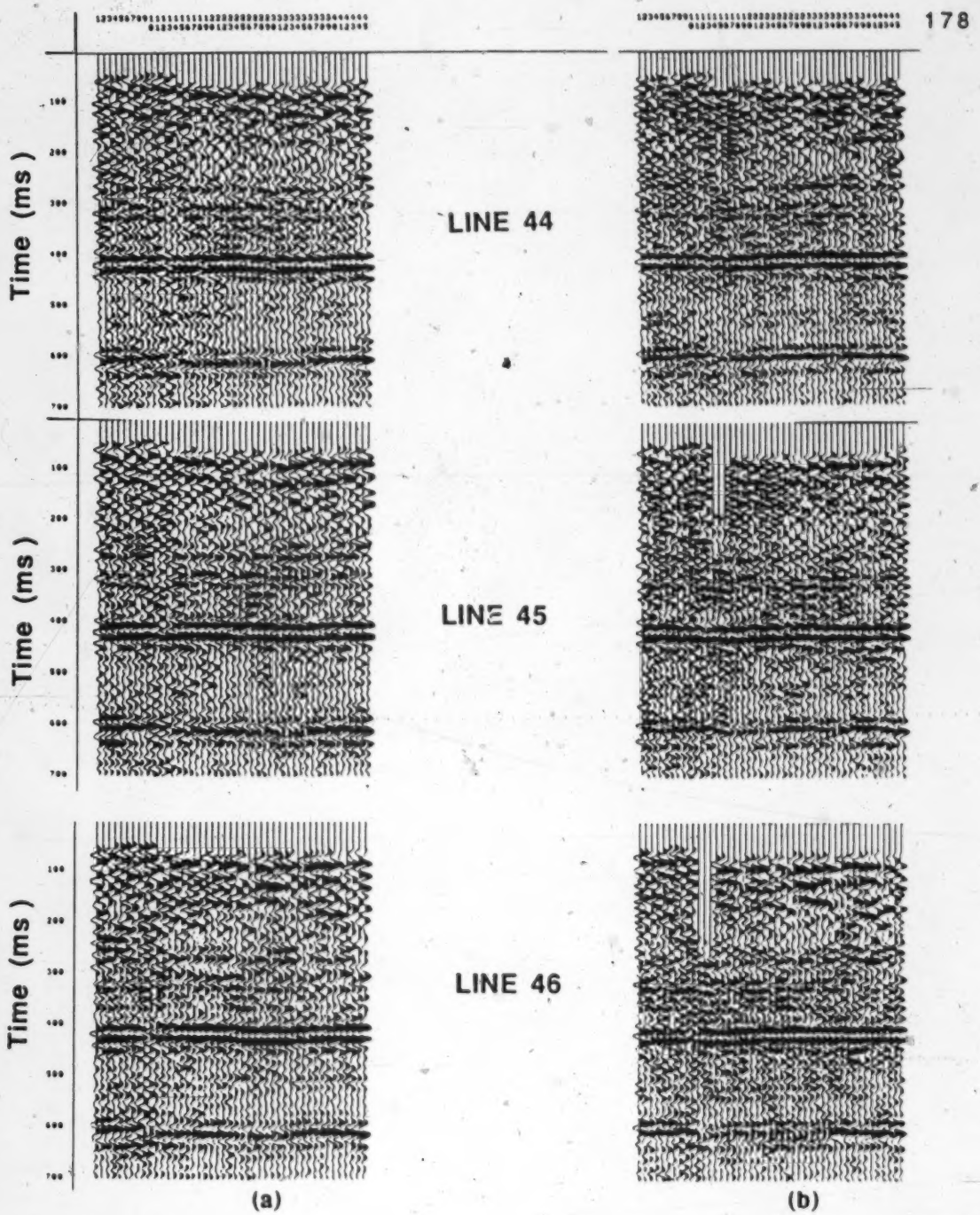


Figure 7.8 Cross section of Common Midpoint stacks for lines 44,45 and 46 before (a) and after (b) steam injection.

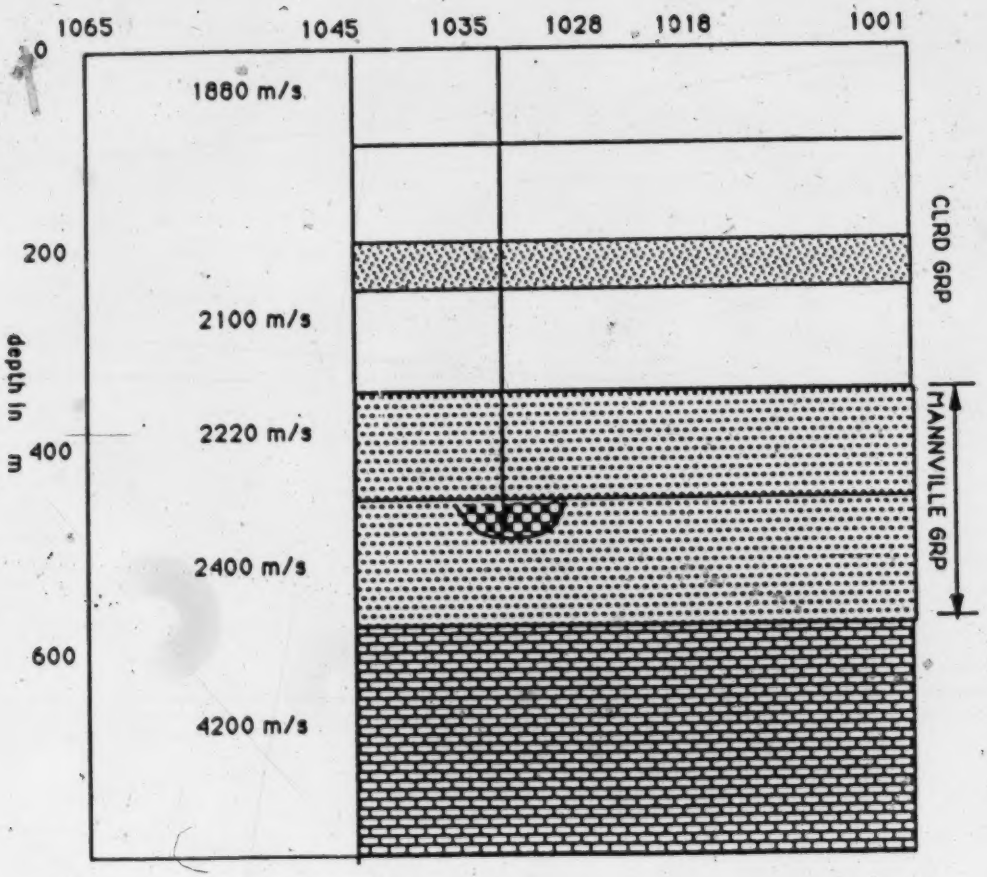


Figure 7.9 Cross section of the velocity-depth model used for 3-D migration.

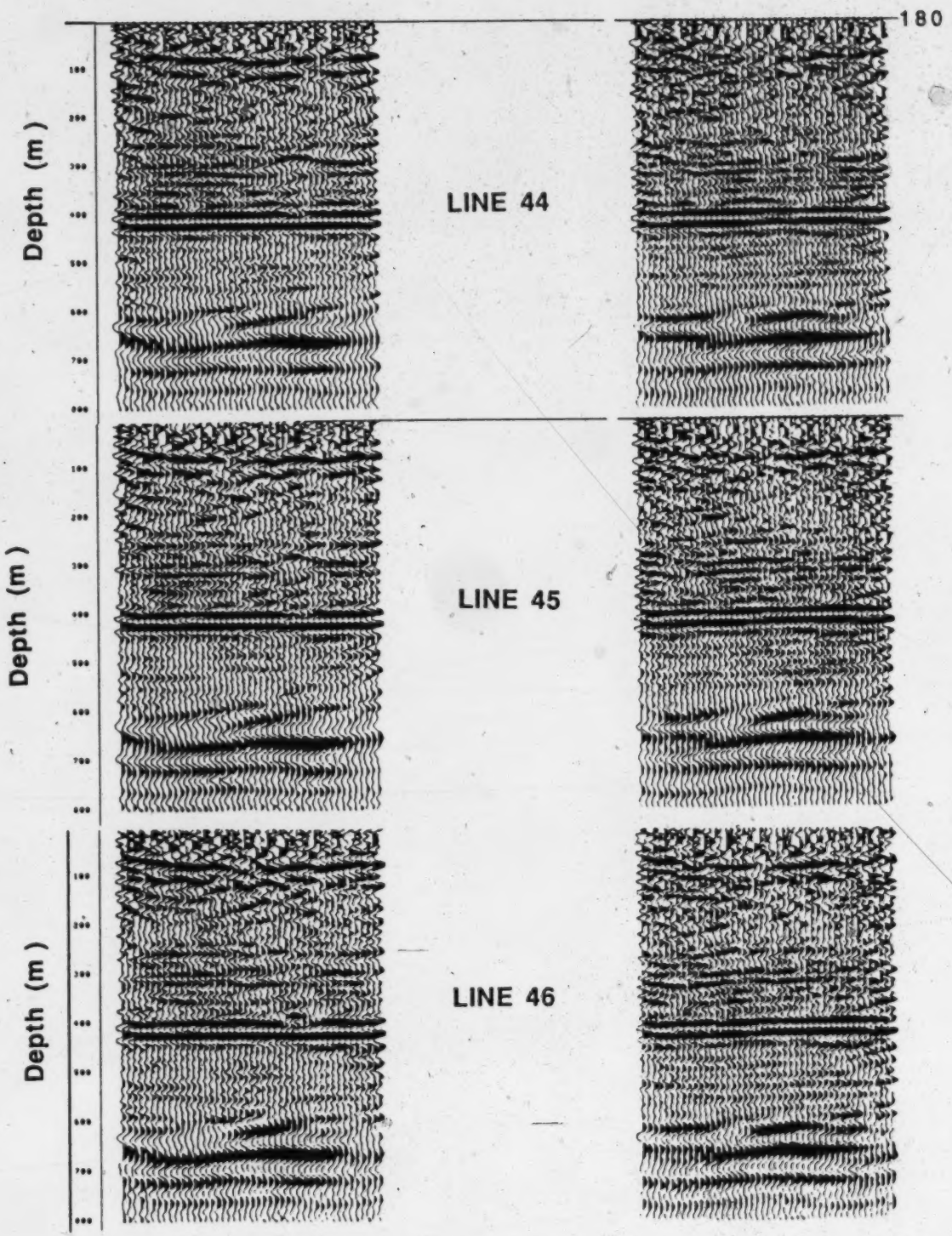


Figure 7.10 3-D migrated cross sections of lines 44,45 and 46 before (a) and after (b) steam injection.

495m). Additional evidence comes from the variations in the Devonian reflector at 600m.

In order to get a better view of the migrated image, as well as to investigate the presence of steam induced sonic anomalies, a 2-D depth slice extending laterally over the region of interest of the migrated data was used. To avoid the influence of unwanted changes in the phase characteristics of the seismic trace four depth slices were summed at depths 480m to 492m, averaged and globally balanced for the *before* and *after* steam sections. Thus, a true amplitude representation of the two sections under comparison was obtained. The color displayed graphs represent wave amplitude variations in the horizontal direction.

The *x-y* depth slices of both sets of data (before-after) were visually scanned using a 3-D orthogonal slicing graphics package on a computer workstation to justify the similarities and differences above and below the steam region. Figure 7.11 shows a set of such planar views at depths of 300m and 600m, from the exact 3-D migration results. The before and after steam images at 300m show the same characteristics as expected. The 600m images are not well correlated possibly because of the influence of steaming on the seismic velocities thus creating a delay in arrival times of the wavefront at the receivers.

Assuming that steam influences the seismic response either directly or because of areal velocity changes due to stress field variations around each well, we turn our attention to the injection level depth planar images. Also, the images obtained by the direct 3-D solution and by the one-pass 3-D with under-relaxation will be compared in order to show which solution is more preferable.

Figure 7.12 shows the location of the wells in the same scale as the 2-D images shown in the latter figures for comparison purposes. The wells to the extreme right of the Figure have gone through a part of second cycle steam injection. Figure 7.13 depicts a 2-D image of the migrated wavefield at depths of 476 to 488 m for before Figure 7.13a and after Figure 7.13b steam injection, using the direct 3-D algorithm. Similarly, Figure 7.14 shows

the same image migrated using the under-relaxed method. First of all we can see that in both cases there is a difference in the amplitude character of the data after steam, especially at the extreme right hand side of the slice where steam was injected for a second cycle. In comparing the migration methods, both solutions seem to point out the fact of steam presence, but there exist some minor differences showing a higher frequency-information content in the exact 3-D case. These can be attributed to the degree of accuracy of the methods, but only if further image processing takes place can this assumption be verified.

In order to examine the effect of steaming, using these image slices I applied an image reconstruction technique and an edge detection process to these migration results. Possibly these methods can reveal more important information at the image viewing level than other conventional approaches such as filtering or averaging. Also, the methods could help distinguish the difference between the two migration algorithms.

The image reconstruction technique used is based on the singular value decomposition (SVD) of a 2-D image matrix and subsequent reconstruction using a desired set of eigenvalues. The method developed by the author under a different research project (Kapotas et al, 1988; Agouridis and Kapotas, 1991), has shown excellent results when applied to 2-D seismic data in any domain. This method will only be used to enhance the dominant characteristics of the depth images. The edge-detection application is based on the implementation in which edges are marked as maxima in gradient magnitude of a Gaussian-smoothed image. The method was described in detail in Chapter 6.

Figures 7.15 and 7.16 show the SVD reconstructed images for the exact and one-pass under-relaxation migrations. An improvement in the overall appearance of the images is evident. The difference between the presteam and poststeam results has become more emphatic, supporting the assumption that the seismic anomalies - related directly or indirectly to steam injection/production - could be imaged by 3-D migration; but their true lateral extent can not be predicted accurately. This is simply a problem of the field

recording parameters used: 25m spacing in the x direction and 100m spacing in the y. Possibly the edge detection approach can partially resolve this problem.

The edge detection results are shown in Figures 7.17 and 7.18. This method should help us determine the lateral extent of the anomaly. Looking at the results carefully one can clearly see the major difference between the before and after steam images. The largest degree of dissimilarity is at the extreme right side of the image indicating the greatest change in velocity structure due to localized steam injection sites. It can be assumed that 3-D migration along with some image-processing tools can localize anomalous zones present in a seismic volume resulting in better delineation of the steam-affected reservoir. It should be pointed out that, throughout this analysis, no direct evidence was accumulated to support the assumption that the observed seismic anomalies present on the data can be attributed to steam injection/production activity in the area.

Finally, in Figure 7.19 a direct comparison of the same depth slice obtained by the exact 3-D and onepass 3-D with under-relaxation is shown. Although some minor differences are present, it can be safely concluded that the onepass 3-D with under-relaxation can be a good and effective approximation of the exact 3-D solution until the next generation of supercomputers can solve the direct 3-D in a reasonable amount of time.

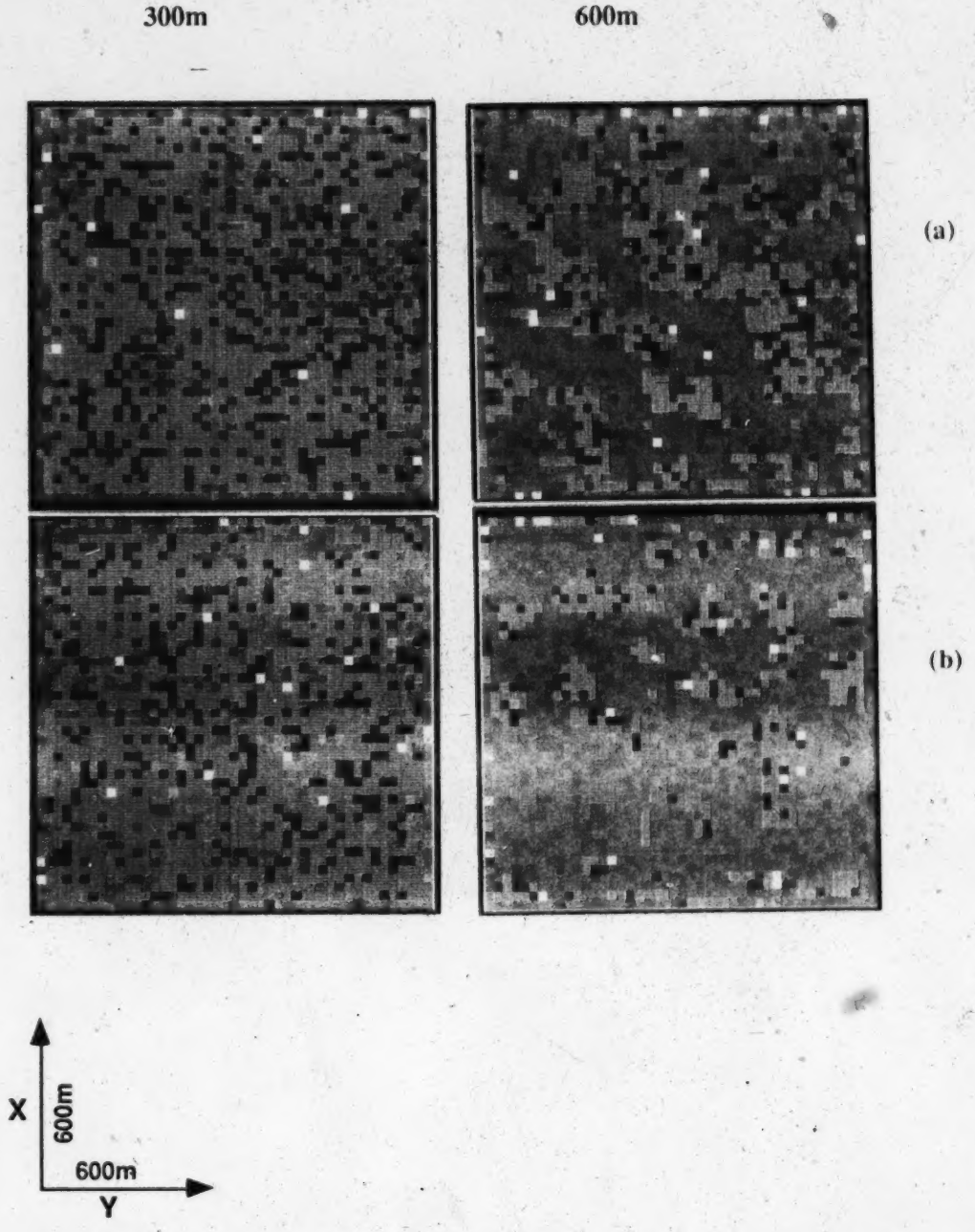
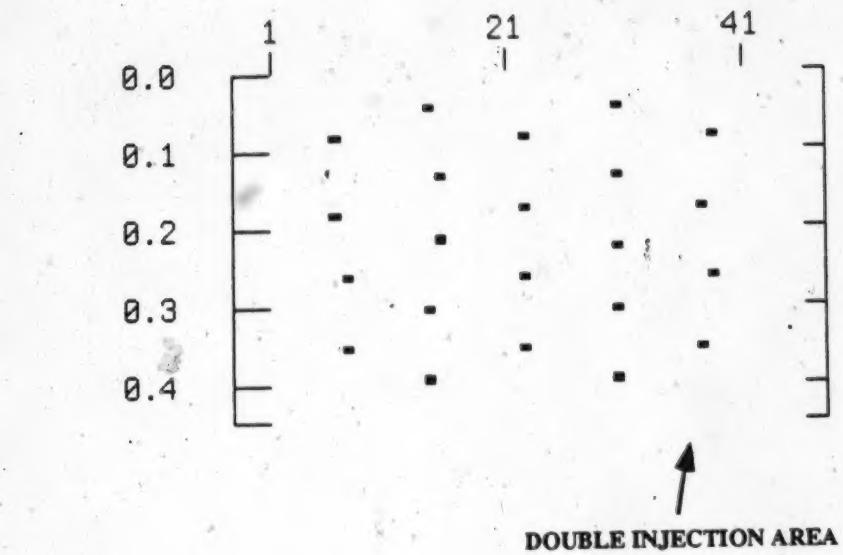


Figure 7.11 Planar view of depth slices at 300m and 600m migrated via the exact 3-D solution.(a) before steam injection (b) after injection.



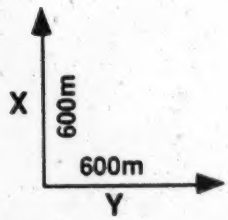


Figure 7.12 Steam injection well locations drawn to scale with the 2-D images from the seismic data for comparison. Inlines and crosslines run from 20 to 65 based on the diagram in Figure 7.7.

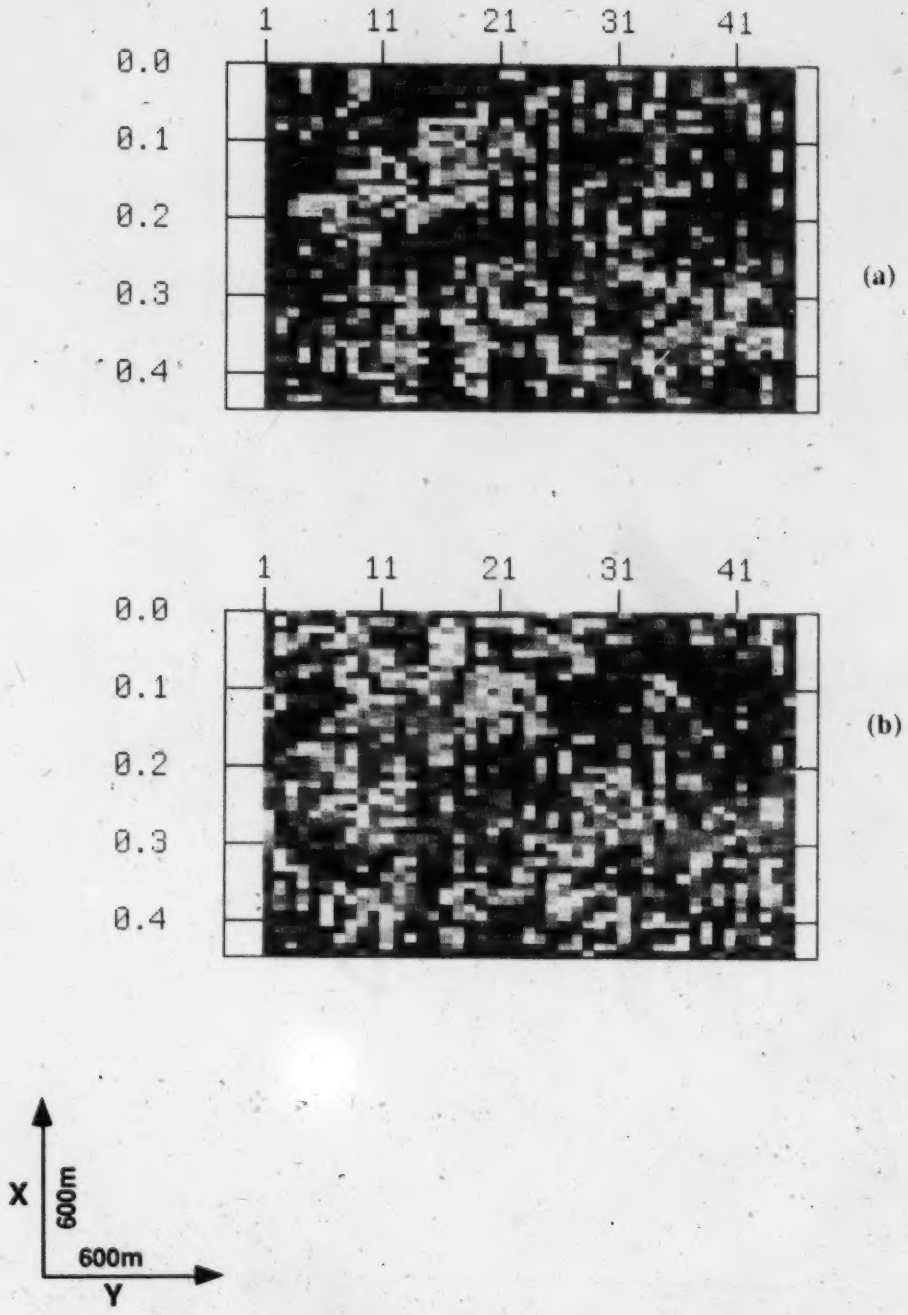


Figure 7.13 2-D depth slices at depths 476 to 488m migrated via the exact 3-D solution.(a) before steam injection (b) after injection.

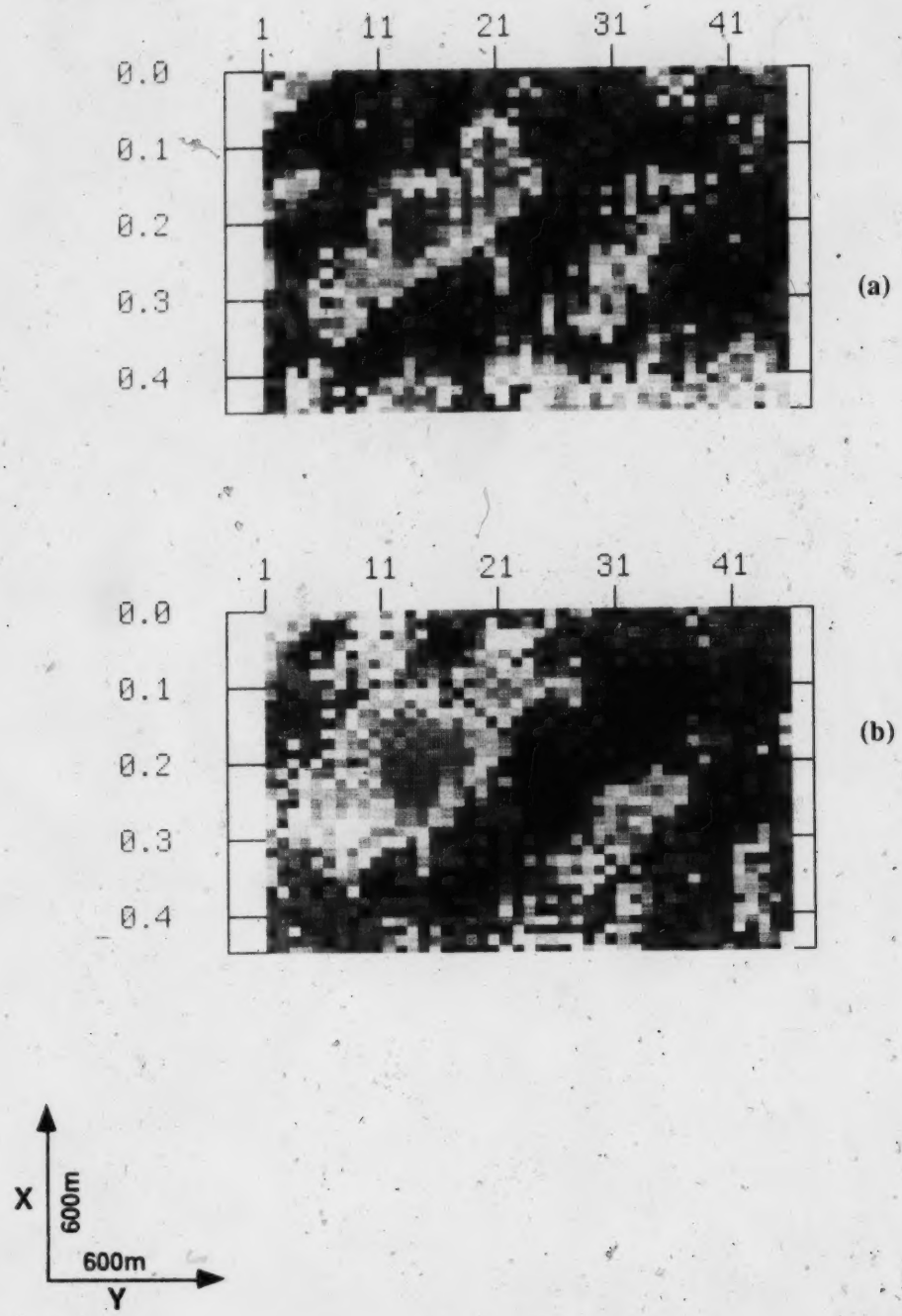


Figure 7.14 2-D depth slices at depths 476 to 488m migrated via the onepass 3-D with under-relaxation solution.(a) before steam injection (b) after injection.

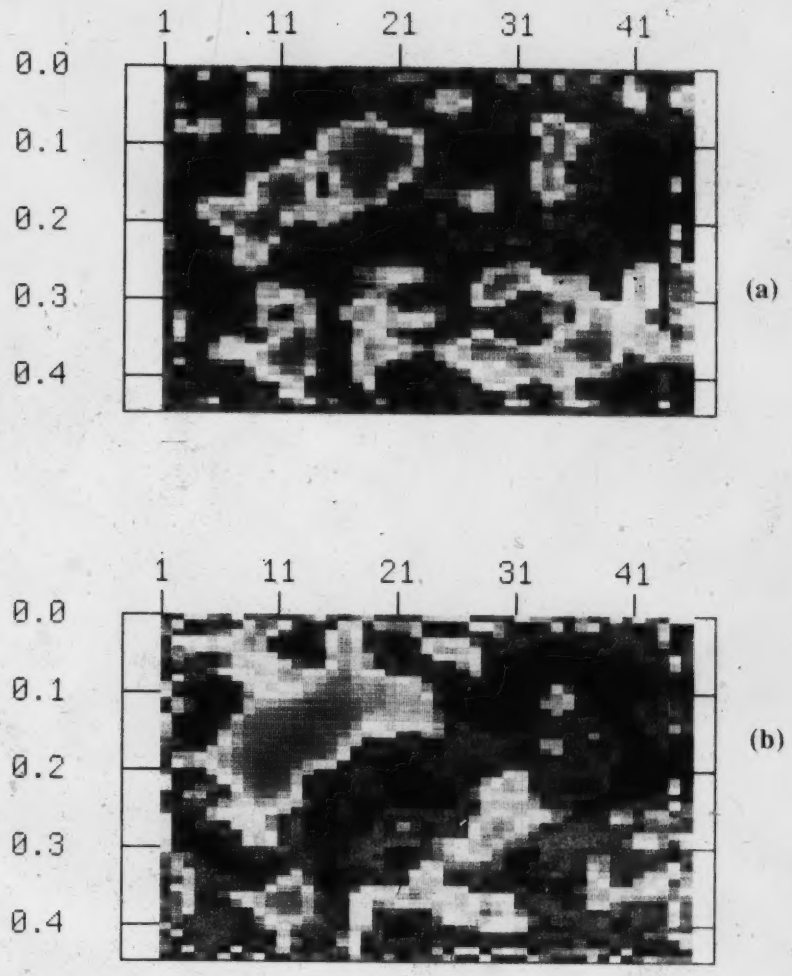


Figure 7.15 2-D depth slices of Figure 6.13 after the SVD enhancement (a)before steam injection (b) after injection.

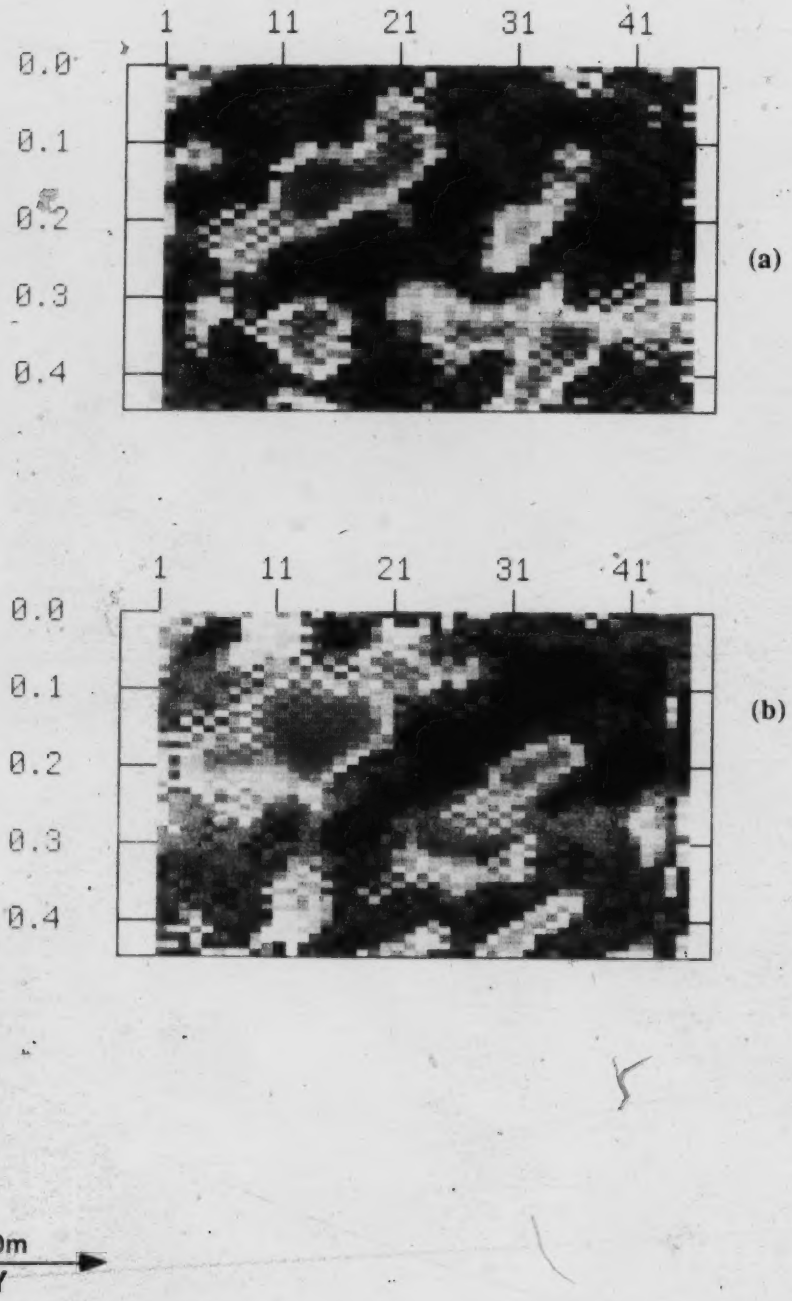


Figure 7.16 2-D depth slices of Figure 7.14 after the SVD enhancement (a)before steam injection (b) after injection.

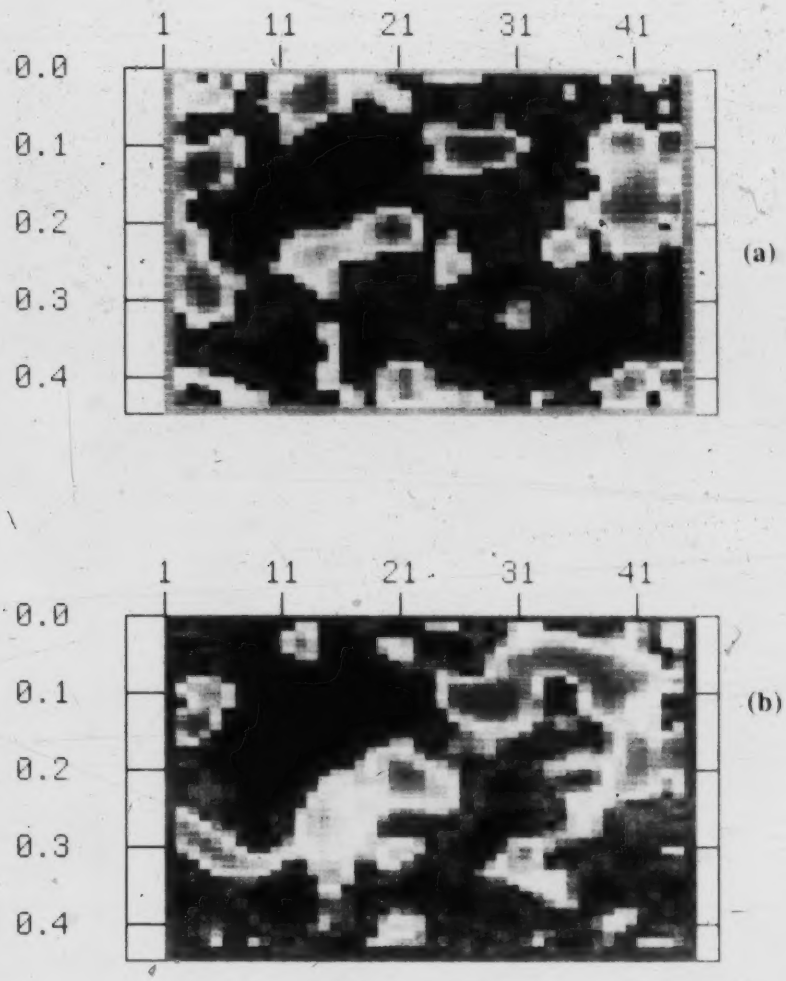


Figure 7.17 2-D depth slices of Figure 7.15 after Edge detection (a)before steam injection (b) after injection.

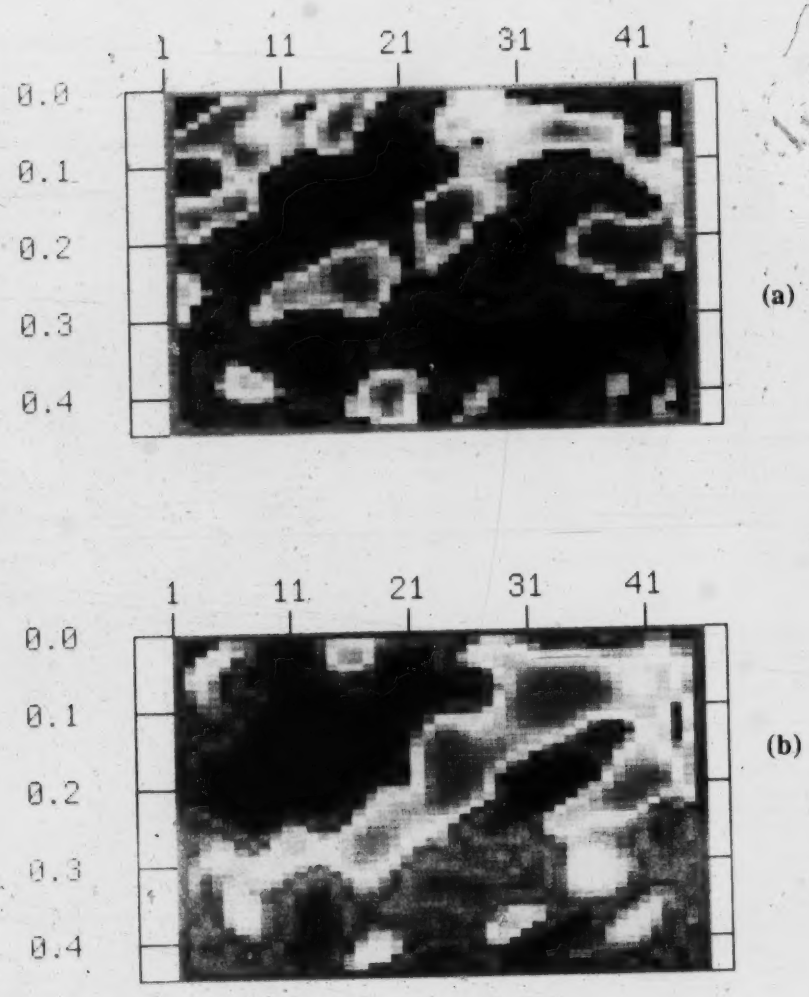
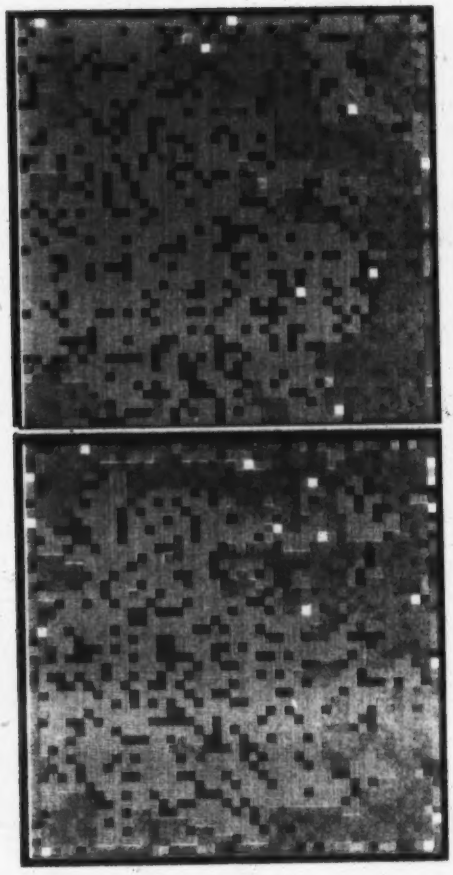


Figure 7.18 2-D depth slices of Figure 7.16 after Edge detection (a)before steam injection (b) after injection.



(a)

(b)

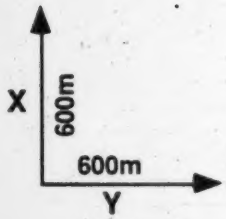


Figure 7.19 Comparison of a depth slice (500m) between (a) exact 3-D migration and (b) onepass 3-D with under-relaxation in order to evaluate the accuracy of the later.

7.3 Conclusion

In this Chapter I have solved the exact 3-D seismic migration problem with the aid of a parallel supercomputer. It has been shown that the method images exactly the subsurface model but it is still very time-consuming on presently available computers. The application of the method to synthetic and real examples leads to the conclusion that it is probably the only alternative in the future for analyzing complex geological regions for the location of hydrocarbon reservoirs.

The extension of the method to the prestack domain based on principles described in Chapter 4 will complete the seismologist's dream: Direct 3-D prestack migration. This part of the research goes beyond the scope of this thesis, but the challenge of the problem will definitely drive me to continue this part of the research under a different project.

CHAPTER 8

EPILOGUE AND PROPHESIES

Recent advances in computing based on a subdivision of the computational sequence into parallel components led me to investigate in this thesis the feasibility of seismic applications on a massively parallel computer. The objective of this research was not only to show dramatic speedups leading to cost reductions in seismic data processing but to allow the implementation of new methods of imaging that were impossible in practice in the past. This thesis described those seismic methods which have been developed and applied into a parallel computer environment in order to investigate their accuracy, speed and efficiency. Emphasis was given to *prestack* migration techniques because of their challenging mathematical formulation in parallel as well as the dynamic improvement they exhibit in imaging the subsurface with seismic reflection data.

The seismic migration problem in terms of the solution of the wave equation in the prestack and poststack domains, in two-dimensional and three dimensional spaces has been derived by means of wavefield extrapolation in the frequency-space domain. In addition to this derivation, the formulation of the three-dimensional acoustic wave forward modeling has been achieved.

Advanced seismic techniques used during this thesis formulation required the computational power delivered by the new generation of parallel computers. Parallel computation is the course of seismic processing leading a more dynamic environment for the development of more accurate methods in the field of seismology. Amongst the wide diversity of parallel computer architectures I used the Myrias SPS line to compile this investigation.

The advent of vector computers allowes the programmer to achieve speedups when the

program is changed to process many multiplication and addition operations sequentially. Similarly, with today's newer breed of parallel computers the ability of the programmer to keep a large number of available processors working is critical to achieving high performance. This research has shown that on some of the currently available parallel machines, high performance can be achieved either through extensive reformulation of the algorithm for a specific class of problems, or with minimal programming changes for other classes.

Prestack depth migration in two-dimensions, in the frequency space domain using the 65-degree approximation of the paraxial wave equation and the principle of double downward continuation was developed and implemented on a parallel computer. Usually such methods are avoided in a typical exploration environment and replaced by others less accurate that run in a much faster time frame. This could lead sometimes to erroneous decisions for the location of hydrocarbons.

The need for more accurate seismic methods implemented in a parallel platform, thus reducing the computational time considerably, led to the development of a prestack depth migration method. Synthetic and real data examples have shown the superiority of the method over similar ones, because the only source of error in this formulation lies in the degree of accuracy of the input velocity model. The parallel formulation of the algorithm has shown performance levels approaching the peak performance level of the hardware used. A similar algorithm for the reverse-time migration problem in the time-space domain did not achieve this level of performance on this platform because of the different formulation of the solution.

It was mentioned earlier that one motivation for using parallel supercomputers in seismic exploration is the ability to process in 3-D instead of 2-D. The existing 3-D work in modeling and migration was constrained by the treatment of velocity as a function of depth only. It was the basis of this research to extend the 3-D problem to include lateral

velocity variations to improve resolution in the results.

An original formulation of the three-dimensional migration problem using *under-relaxation* for the solution of a pseudo-tridiagonal system of equations reduced the computational time needed to solve the system directly by a factor of ten. Applications of the method proved the superiority of three dimensional processing over two-dimensional one. Three-dimensional migration analysis was applied to a field data set over a steam injection region in Cold Lake heavy oil recovery area. Results show the change in reflection characteristics due to steam presence when compared with before steam injection data. Improvements of the *under-relaxation* method in the future along with extensions to the solution for the 45 degree approximation will benefit the seismic exploration world.

An innovative extension of the three dimensional migration to the prestack domain proved that parallel computing power, not only in terms of floating point operation but also in system memory capabilities, can be used efficiently to accomplish this massive operation. Synthetic examples, generated using 3-D acoustic modeling, justified the accuracy of the algorithm and showed the superiority of three dimensional prestack seismic imaging.

The 3-D acoustic modeling mentioned above, was developed in this thesis using principles of finite-difference solutions for differential equations. A stable scheme with a dynamic set of boundary conditions was formulated and implemented in parallel. The need of 3-D modeling, not only to test the 3-D migration, comes from the necessity of high resolution simulations using smaller grid spacings and larger models to understand the complexity of observed seismic records. It is obvious that the memory limitations of some vector computers confined algorithms to small models, but now the parallel analogue that provides hundreds of megabytes of system memory can help us overcome this problem.

The basic mathematical formulation and solution of the direct three dimensional

migration in the 45 degree approximation unfolded horizons that lead to higher degrees of resolution and accuracy in the processing of seismic data. Applications of the method have shown the degree of superiority over other pseudo three dimensional solutions, but the tradeoff in computer time still makes this method difficult to use. Improvements in the solution of complex sparse matrices as well as increase in the performance of parallel computers could justify the use of direct solutions to the three dimensional migration problem. Extension of the method to the prestack case can be easily carried out using principles described in this research work.

Modifications of the two dimensional and three dimensional algorithms to accommodate anisotropy can easily be accomplished and used to improve the imaging of seismic events. Phadke et al (1991) have shown the implementation of anisotropy in the 2-D poststack migration algorithm in parallel thus paving the way for similar extensions to 3-D.

In this dissertation a complete theoretical and numerical exposition of the three dimensional prestack and poststack seismic migration and acoustic wave propagation has been presented. Synthetic and real examples were used as part of the analysis emphasizing the need of three dimensional seismic processing. The methods have shown to be practical with the availability of parallel supercomputers. There is still much to be learned in applying high-performance parallel processors in exploration geophysics. Future phases of this investigation could utilize this next generation of supercomputers and extend the direct (exact) three dimensional migration to prestack space, including anisotropy and possibly elastic wave behaviour. The usefulness of these architectures in seismic work will continue to play an important role in quest for more sources of energy.

BIBLIOGRAPHY

Agouridis, D. and Kapotas, S., 1991, Enhancement of Seismic data via the SVD Transform, Proceedings Canadian Geophysical Union, National Convention, May 1-4, 1991, Banff, Canada.

Aki, K. and Richards P. G, 1980, Quantitative Seismology, Theory and methods, W. H Freeman and Co., San Francisco.

Alford, R., and Kelly, K. 1974, Accuracy of finite-difference modeling of the acoustic wave equation, Geophysics, **39,** 834-842.

Baysal, E., Kosloff, D.D., Sherwood, J.W., 1983, Reverse time migration, Geophysics, **48,** 1514-1524.

Berkhout, A. J., 1987, Applied seismic wave theory, Elsevier Science Publishers, N.Y.

Berkhout, A. J., 1982, Seismic Migration, Elsevier Science Publishers, Amsterdam.

Berkhout, A. J., 1981, Wave field extrapolation techniques in seismic migration, a tutorial. Geophysics, **46,** 1638-1656.

Black, J.L. and Leong, T.K., 1987, A flexible, accurate approach to 1-pass 3d Migration, 57th annual SEG Int. Meeting, New Orleans, October 1987.

Boore, D., 1972, Finite-difference methods for seismic wave propagation in heterogeneous materials, Methods of Computational Physics, **11,** Academic Press, 1-37.

Brown, D.L., 1983, Applications of operator separation in reflection seismology, Geophysics, **48,** 288-294.

Canny, J., 1986, A computational approach to edge detection, IEEE transactions on pattern analysis and machine intelligence, PAMI-8, **6,** 679-698.

Carter,J.A. and Frazer,N.L., 1984, Accommodating lateral velocity changes in Kirchhoff migration by means of Fermat's principle, Geophysics, **49,** 46 - 53.

Cary, P.W., 1990, Symmetric one-pass 3-D migration, CSEG Recorder, **10,** 3-15.

Chang, W. F., and McMechan G. A., 1986, Reverse - time migration of offset vertical seismic profiling data using the excitation time imaging condition, Geophysics, **51,** 67-84.

Claerbout, J.F., 1970, Coarse grid calculations of waves in inhomogeneous media with application to delineation of complicated seismic structure. Geophysics, **35,** 407-418.

Claerbout, J.F., 1976, Fundamentals of Geophysical data Processing, McGraw Hill Inc.

Claerbout, J.F., 1985, Imaging the Earth's Interior, Blackwell Scientific Publications.

Clark, J.J., 1989, Authenticating edges produced by zero-crossing algorithms, IEEE transactions on pattern analysis and machine intelligence, PAMS-2 , **1,** 43-57.

Clayton R.W. and Engquist B., 1980, Absorbing boundary conditions for wave equation migration, Geophysics, **45,** 895-904.

Clayton R.W. and Engquist B., 1977, Absorbing boundary conditions for acoustic and elastic wave equations, Bull. Seism. Soc. Am., **67,** 1529 - 1540.

Clifton, R.J., 1967, A difference method for plane wave problems in elasticity. Quart. Appl. Math., **25,** 97-116.

DeGolyer, E. , 1935, Notes on the early history of applied Geophysics in the petroleum industry. Journal of the Society of Petroleum Geophysicists, **6,** 1-10.

Dickinson, J.A., 1988, Evaluation of two-pass three-dimensional migration, Geophysics, **53,** 32-49.

Duncan, R.,1990, A survey of parallel computer architectures, Computer, I.E.E.E., **2,** 5-16.

Eisner, E., 1989, Supercomputers in seismic exploration. Handbook of Geophysical Exploration, Editors K. Helbig, S. Treitel, Pergammon Press. N.J.

Flynn, M., 1966, Very high Speed Computing Systems, Proc. IEEE, Vol. **54,** pp. 1901-1909

Fox, G., 1989, Solving problems on Concurrent processors., Volume I, Prentice Hall, N.J.

French, W., 1975 Computer migration of oblique seismic reflection profiles, Geophysics, **40,** 961-980.

Fricke, J.R., 1988, Reverse time migration in parallel: a tutorial. Geophysics, **53,** 1143-1150

Froidevaux, P., 1990, First results of a 3-D Prestack migration program, S.E.G. Expanded Abstracts, 60th Convention San Francisco, **II,** 1318-1321.

Gazdag, J., 1978, Wave-equation migration with the phase-shift method. Geophysics, **43,** 1342-1351.

Gibson, B., Larner, K., and Levin, S., 1983, Efficient 3-D migration in two steps, Geophysical Prospecting, **31,** 1-33.

Golub., G.H. and Van Loan, C.F. 1898 Matrix computations, The John Hopkins University Press, Baltimore.

Hague, J. and Goloway, F., 1989 3-D omega-x migration on the IBM 3090/VF, Supercomputer, **33,** VI-5, 17-27.

Hale, D. 1990, 3-D Depth migration via McClellan Transformations, S.E.G. Expanded Abstracts, 60th Convention San Francisco, **II,** 1325-1328.

Hildebrand, F.B. 1974, Introduction to Numerical Analysis, New York, McGraw-Hill.

Huertas, A., Medioni G., 1986, Detection of intensity changes with subpixel accuracy using Laplacian-Gaussian Masks, IEEE transactions on pattern analysis and machine intelligence, PAMI-8, **5,** 651-664.

Hwang, K., 1984, Tutorial supercomputers: Design and Applications, Computer Society Press, Los Alamitos, Calif.

Israeli, M.and Orszag, S.A., 1981, Approximation of radiation boundary conditions, Journal of Computational Physics, **41,** 115-135.

Jain, S., and Wren, A. E., 1980, Migration before stack - Procedure and significance, Geophysics, **45,** 204 - 212.

Jakubowicz, H. and Levin, S. 1983, A simple exact method of 3-D migration, Geophysical Prospecting, **31,** 34-56.

Joe, B., 1990, Parallel Sparse LU Factorization on the Myrias SPS-2 Computer, Internal Report Myrias Research Corporation.

Johnston, R. L., 1982, TEAPACK user's manual mathematical subroutines for numerical methods, John Wiley & Sons, Inc.

Jonson, O., 1984 Three Dimensional Wave equation Computations on Vector Computers. Proceeding of IEEE, **72,** 90-95.

Julien, P.,1990, 3-D Prestack Depth migration on Real Data, S.E.G. Expanded Abstracts, 60th Convention San Francisco, **II,**1329-1332.

Kanasewich, E. R. and S. M. Phadke, 1990, Ray tracing in inhomogeneous media, A.O.S.T.R.A. report.

Kanasewich, E. R. and S. M. Phadke, 1988, Imaging discontinuities on seismic sections, Geophysics, **53,** 334-345.

Kanasewich, E. R., 1968, Precambrian rift: genesis of strata-bound ore deposits: Science, **161,** 1002-1005.

Kapotas, S., Kanasewich, E.R. and Phadke, S., 1990b, Seismic Imaging with a Parallel MIMD computer, Proceedings Society of Exploration Geophysics, National Convention, September 23-27, 1990, San Francisco, U.S.A., Vol **I,** 347-350.

Kapotas, S., Kanasewich, E. R., and Phadke S., 1990a, Advancing into the 90's: Parallel seismic applications on the SPS-2 . Presented C.S.E.G. '90, Calgary, Alberta.

Kapotas, S. and Kanasewich, E.R., 1990b, Seismic Imaging on the SPS-2 parallel computer, Proceedings Supercomputing Symposium 90, Ecole Polytechnique, Montreal, June 4-6 1990, Vol. **4**, pg. 45-54.

Kapotas, S. and Kanasewich, E.R. 1990a, Parallel seismic applications on the Myrias SPS-2 . Presented SPAA'90, Crete, Greece.

Kapotas, S., Kanasewich, 1989, Seismic applications on the Myrias SPS-2 . Invited paper, Supecomputing '89, Geophysical workshop, Reno, Nevada.

Kapotas, S., Agouridis, and D., Nguyen, D. 1988, A new image reconstruction technique for enhancement of seismic data, Proceedings Society of Exploration Geophysics, National Convention, November 1-4, 1988, Anahaim, U.S.A., Vol **II**, pg. 1325-28.

Kelamis, P.G., and Kjartansson, E., 1985, Forward modeling in the frequency space domain, Geophys. Prosp., **32**, 252 - 262.

Kelly, K.R., Ward, R.W., Treitel, S, Alford, R. M., 1976, Finite-difference synthetic seismograms, Geophysics, **41**, 2-27.

Korn, M. and Stoekel, H., 1982, Reflection and transmission of Love channel waves at coal seam discontinuities with a finite-difference method, Journal of Geophysics, **50**, 1771-1788

Kosloff D. , and Baysal, E., 1983, Migration with the full wave equation, Geophysics **48**, 677 - 687.

Lacroix, V., 1989, A three-module strategy for edge detection, IEEE transactions on pattern analysis and machine intelligence, PAMS-10 , **6**, 803-810.

Lowenthal, R., Roverson, L., Lu, R.and Sherwood, J., 1976, The wave equation applied to migration. Geophysical Prospecting, **24**, 380-399.

Marr D. and Hildreth E., 1980, Theory of edge detection, Proc. Roy. Soc. London, vol. **B207**, 187-217.

McDermott, E., 1932, Application of Reflection Seismograph. Transactions of the Society of Petroleum Geophysicists, **III**, 34-39.

McMechan, G. A. 1983, Migration by extrapolation of time dependent boundary values, Geophysical Prospecting, **31**, 413-420.

McMechan, G. A., Mooney, W. D., 1980, Asymptotic ray theory and synthetic seismograms for laterally varying structures: Theory and application to the Imperial Valley, California, Bull. Seis. Soc. Am. **70**, 2021-2035

Mitchell, A.R., 1969, Computational methods in partial differential equations: New York, John Wiley and Sons.

Mitchell, A.R., Griffiths, D.F., 1980, The finite difference method in Partial Differential Equations, John Wiley & Sons.

Mora, P. 1988, 3-D acoustic forward modeling in the CM2, Presented at the SEG Int. Meeting, Anahaim, Calif., Oct. 1988.

Mufti, I. R., 1989, Application of supercomputers in three dimensional seismic modeling. In Supercomputers in Seismic Exploration, Editors, Helbig and Treitel, Pergammon Press, N.Y.

Muller, G, Temme, P., 1987, Fast frequency - wavenumber migration for depth dependent velocity, Geophysics, **52,** 1483 - 1491.

Myczkowski, J., McCowan, D., Mufti, I., 1991, Finite-difference seismic modeling in real time. SEG, Leading Edge, June 1991, 49-52.

Myrias 1990, PAMS 2.3.4., The Myrias SPS system, Internal Report, Edmonton, Alberta.

Phadke, S., Kapotas, S. and Kanasewich E.R., 1991, Migration of P-wave reflection data in transversely isotropic media. Geophysics, (To be submitted)

Phadke, S., Kapotas, S. and Kanasewich E.R., 1990,. Ray tracing on the Myrias SPS-2, Internal Report, Physics Department, University of Alberta.

Raynolds, A.C., 1978, Boundary conditions for the numerical solution of wave equation propagation problems, Geophysics, **43,** 1099-1110.

Reshef, M., Kosloff, D., Edwards, M.,Hsiung, C., 1988, Three-dimensional acoustic modeling by the Fourier method, Geophysics, **53,** 1175-1183.

Reshef, M., Kosloff, D., 1986, Migration of common shot gathers, Geophysics, **51,** 324 - 331.

Ristow, D. , 1980, 3-D finite difference migration, Ph.D. thesis, University of Utrecht, The Netherlands.

Santosa, F., and Pao, Y.H., 1986, Accuracy of a Lax-Wendroff scheme for the acoustic wave equation. J. Acoustic Soc. Am., **80,** 5, 1429-1437.

Sattleger, J.W., Stiller, P.K., 1974, Section migration, before stack, after stack, or in between. Geophysical Prospecting, **22,** 297-314.

Schneider, W.A., 1978, Integral formulation for migration in two and three dimensions, Geophysics, **43,** 49-76.

Schultz, P.S.,and Sherwood, J.W.C., 1980, Depth migration before stack, Geophysics, **45,** 376 - 393.

Smith, G.D., 1965, Numerical Solution to Partial Differential Equations, Oxford University Press, London..

Stoffa, P., Fokkema, J., Freire, L.R., Kessinger, W.P., 1990, Split-step Fourier migration. Geophysics, **55,** 410-421.

Stolfo, S., and Miranker D., 1986, The DADO production system machine, J. Parallel and Distributed Computing, Vol. **3,** No.2.

Stolt, R., 1978, Migration by fourier transform. Geophysics, **43,** 23-48.

Stolt, R.,and Benson, A.K.,1986, Seismic migration - theory and practice, Geophysical Press.

Stone, T, Joe, B., Behie, A., London, M., 1990, Solution of linear systems on a MIMD parallel computer, Internal Report Myrias Research Corporation.

Stone, T., Kapotas, S., Malcom, D., Petherick, L., 1990, Parallel computing for oil industry applications, Submitted for publication, Energy Processing Canada.

Sun, R. and McMechan G., 1986, Pre-stack reverse time migration for elastic waves with application to VSP profiles. Proc. IEEE, vol. **74,** No.3, 457-465.

Temme, P., 1984, A comparison of common midpoint, single-shot and plane wave depth migration, Geophysics, **49,** 1896 - 1907.

Thorson, J., 1980, Synthetic examples of prestack migration, Stanford Exploration Project Report No. 24, Stanford, California, pp. 5-32.

Tore, V., Poggio, T., 1986, On Edge Detection, IEEE transactions on pattern analysis and machine intelligence, PAMI-8, **2,** 147-163.

Tsingas K., 1990, Seismic reflection imaging, PhD dissertation, University of Alberta, Edmonton.

Vafidis, A., 1988, Super computer finite difference methods for seismic wave propagation, PhD dissertation, University of Alberta, Edmonton.

Van Doormaal J., 1990, The GAS3D solution method, Internal report C. Engineering Department, BYU.

Wapenaar, C.P.A., Kinneging, N.A., and Berkhout, A.J.,1987, Principle of prestack migration based on the full elastic two-way wave equation, Geophysics, **52,** 151 - 173.

Weatherby, B. B., 1940, The history and development of seismic prospecting. Geophysics , **5**, 215-230.

Yilmaz, O, 1987, Seismic Data Processing, Society of Exploration Geophysics, Tulsa OK.

Yilmaz, O., Claerbout, J. F.,1985, Prestack partial migration, Geophysics, **45**, 1753 - 1779.

APPENDIX A

PARAXIAL (PARABOLIC) WAVE EQUATION.

Assume that the reflected compressional waves at receivers not too far from the source can be represented by the following scalar wave equation:

$$\frac{\partial^2}{\partial x^2} P(x,z,t) + \frac{\partial^2}{\partial z^2} P(x,z,t) = \frac{1}{v^2(x,z)} \frac{\partial^2}{\partial t^2} P(x,z,t) \qquad \text{(A1)}$$

This will be true for a solid earth if the converted compressional to shear wave energy is negligible and this occurs physically when the waves are propagating close to vertical incidence (angle of incidence $< 20^0$). This equation will also be true for pure horizontally polarized (SH) waves or for waves recorded in water in a marine survey.

Another approximation used very frequently in exploration geophysics is to assume that the source and receiver are in identical locations on the surface and the waves are true echoes similar to that used with electromagnetic waves or radar. To make this approximation more exact one corrects for the extra travel time for spatially separated source and receivers and derives a new data set located midway between source and receiver, called the Common midpoint (CMP) or Common depth point (CDP) trace. This trace can be regarded as due to an upcoming wave propagating from the reflecting surface to the CMP receiver at half the actual wave velocity. This formulation is called the "exploding reflector model", first introduced by Claerbout (1976), because it appears as if the recorded echoes P(x,0,t), or reflections are due to an explosion at the reflecting beds at time t=0. The problem is then one of extrapolating the wavefield to depth z, and solving for the field at P(x,z,0).

In terms of computational aspects and also parallel task distribution it is advantageous to decompose the wavefield into monochromatic plane waves with different angles of propagation from the vertical. This is also an approximation but a good one in most instances because the radius of curvature of the wavefront is very large (usually > 1000m) compared to the spatial separation of sources and receivers (< 50m). Therefore, we work in the Fourier transform domain. A Fourier transform is taken over time or space (or sometimes both) in various extrapolation procedures.

In principle such a decomposition will require appropriate modifications to the scalar wave equation in order to accommodate the behavior of the Fourier transformed wavefield. This observed field can always be temporally transformed into the frequency domain. In cases where there are small lateral velocity variations, one can also transform over the horizontal axis x. Let us look at this case first and then analyze the situation where lateral velocity variations exist.

The Fourier transformed field is given by:

$$P(k_x, z, \omega) = \int \int P(x, z, t)\, e^{(ik_x x - i\omega t)} dx dt \qquad \text{(A2)}$$

and inversely,

$$P(x, z, t) = \int \int P(k_x, z, \omega) e^{(-ik_x x + i\omega t)} dk_x d\omega \qquad \text{(A3)}$$

One must assume that the velocity (usually a root mean square average) is independent of the z spatial coordinate. Apply the operator in equation (2.5) to the scalar wave equation to get:

$$\frac{\partial^2 P}{\partial z^2} + \left[\frac{\omega^2}{v^2(x)} - k_x^2\right] P = 0 \qquad \text{(A4)}$$

The solution to this equation is of the form:

$$P(k_x, z, \omega) = P(k_x, 0, \omega) \exp\left(-i\left[\frac{\omega^2}{v^2} - k_x^2\right]^{1/2} z\right) \qquad \text{(A5)}$$

which is also a solution of the one-way scalar wave equation:

$$\frac{\partial P}{\partial z} = i\frac{\omega}{v}\sqrt{1 - \frac{v^2(x)\ k_x^2}{\omega^2}}\ P \qquad \text{(A6)}$$

Let us define:

$$k_z(z) = \pm\frac{\omega}{v}\sqrt{1 - \frac{v^2(x)\ k_x^2}{\omega^2}} \qquad \text{(A7)}$$

then equation (A5) becomes,

$$P(k_x, z, \omega) = P(k_x, 0, \omega) \exp\left(-i\int_0^z k_z(z)\,dz\right) \qquad \text{(A8)}$$

This formulation can be used to extrapolate the field from z=0 to z by a phase shift term. Because of the monochromatic wave independency of this equation, its solution on an MIMD parallel computer is dynamic and efficient during extrapolation to the desired depth level z.

APPENDIX B

F-D FORMULATION OF THE 65^0 2-D PARAXIAL WAVE EQUATION

In order to compute the monochromatic wavefield at Q(x,z), we discretize the differential equation (3.18a) -(3.18b can be solved the same way), and approximate Q(x,z) by a mesh function q_n^m where m=i Δx , with i=1,.....,Nx and n=k Δz , with k=1,....,Nz. Δx and Δz are the finite difference mesh grid spacings for the horizontal and vertical axes respectively. Nx indicates the number of traces in the gather and Nz the number of depth steps.

Replacing the differentials of equation (3.18) by their respective finite differences and changing the sign of the z differentials to evaluate upgoing waves one has:

$$-\frac{i\beta}{2\alpha\,\mu}\,\delta_{xx}\delta_z\,Q-\frac{\mu\,i}{\alpha}\,\delta_z\,Q+\delta_{xx}\,Q=0 \qquad \text{(B1)}$$

where $\delta_z = \partial/\partial z$, $\delta_{xx} = \partial^2 / \partial x^2$, and $\mu(x,z)=\omega/v(x,z)$.

A higher order (fourth) of accuracy for the approximation of the second order differential is the Douglas formula and is given by (Mitchell and Griffiths, 1980; Claerbout, 1985) :

$$\delta_{xx}=\frac{1}{\Delta x^2}\left(\frac{\delta^2}{1+\gamma\delta^2}\right) \qquad \text{(B2)}$$

where Δx is the grid spacing and $\gamma = 1/12$ (Mitchell and Griffiths (1980) or 1/6 Claerbout (1985), and δ^2 is the operator (1, -2, 1).

An implicit Crank-Nicolson (Mitchell and Griffiths, 1980; Claerbout 1976,1985) finite difference algorithm scheme is employed to solve the resultant difference equation. This algorithm is unconditionally stable (Mitchell and Griffiths, 1980) and involves more than one point at the advanced depth level according to :

$$q_n^m = \frac{1}{2}\left(q_{n+1}^m + q_n^m \right) \tag{B3}$$

The finite difference approximation of the differential equation (B1) employing equations (B2) and (B3) reads:

$$-\frac{i}{2\,\alpha\Delta x^2 \mu}\left(\frac{\delta^2}{1+\gamma\delta^2}\right)\left(\frac{q_{n+1}^m - q_n^m}{\Delta z}\right) + \frac{1}{2}\frac{\delta^2}{\Delta x^2(1+\gamma\delta^2)} q_{n+1}^m + \\ + \frac{1}{2}\frac{\delta^2}{\Delta x^2(1+\gamma\delta^2)} q_n^m - \frac{\mu\beta\, i}{\alpha\Delta z}\left(q_{n+1}^m - q_n^m\right) = 0 \tag{B4}$$

Multiply throughout by $\Delta x^2 (1+ \gamma\delta^2)$ we have:

$$-\frac{i}{2\alpha\,\mu}\frac{\delta^2}{\Delta z}\left(q_{n+1}^m - q_n^m\right) + \frac{\delta^2}{2}\left(q_{n+1}^m + q_n^m\right) - \frac{\mu\, i\beta\, \Delta x^2}{\alpha\Delta z}(1+\gamma\delta^2)\left(q_{n+1}^m - q_n^m\right) = 0 \tag{B5}$$

Simplifying by setting the laterally varying coefficients as A and B one has:

$$A = \frac{i}{2\alpha\,\mu\,\Delta z}$$

$$B = \frac{\mu\, i\,\beta\Delta x^2}{\alpha\Delta z}$$

and apply the operator $\delta^2_{\cdot} = (1, -2, 1)$ we have:

$$
\begin{aligned}
&- A\, q_{n+1}^{m+1} + 2A q_{n+1}^{m} - A\, q_{n+1}^{m-1} + A\, q_{n}^{m+1} - 2A q_{n}^{m} + A\, q_{n}^{m-1} \\
&+ \frac{1}{2} q_{n+1}^{m+1} - q_{n+1}^{m} + \frac{1}{2} q_{n+1}^{m-1} + \frac{1}{2} q_{n}^{m+1} - q_{n}^{m} + \frac{1}{2} q_{n}^{m-1} \\
&- B q_{n+1}^{m} + B q_{n}^{m} - B\gamma\, q_{n+1}^{m+1} + 2B\gamma\, q_{n+1}^{m} - B\gamma\, q_{n+1}^{m-1} \\
&+ B\gamma\, q_{n}^{m+1} - 2B\gamma\, q_{n}^{m} + B\gamma\, q_{n}^{m-1} = 0
\end{aligned}
\qquad \text{(B6)}
$$

Separate the terms with index n+1 to the left hand side and the terms with index n to the right hand side and after some algebraic simplifications we have:

$$
\begin{aligned}
&\left(-A + \frac{1}{2} - B\gamma\right) q_{n+1}^{m+1} + (2A - 1 - B + 2B\gamma) q_{n+1}^{m} + \\
&+ \left(-A + \frac{1}{2} - B\gamma\right) q_{n+1}^{m-1} = \\
&\left(-A \frac{1}{2} - B\gamma\right) q_{n}^{m+1} + (2A + 1 - B + 2B\gamma) q_{n}^{m} + \\
&+ \left(-A \frac{1}{2} - B\gamma\right) q_{n}^{m-1}
\end{aligned}
\qquad \text{(B7)}
$$

Replacing the coefficients of the q's and note that the first is equal to the third we get for the left hand side:

$$\overline{A} = \left(-A + \frac{1}{2} - B\gamma\right)$$

$$\overline{B} = (2A - 1 - B + 2B\gamma)$$

and for the right hand side:

$$\overline{C} = (-A - \frac{1}{2} - B\gamma)$$

$$\overline{D} = (2A + 1 - B + 2B\gamma)$$

or in compact form:

$$\overline{A}q_{n+1}^{m+1} + \overline{B}q_{n+1}^{m} + \overline{A}q_{n+1}^{m-1} = \overline{C}q_{n}^{m+1} + \overline{D}q_{n}^{m} + \overline{C}q_{n}^{m-1}$$

(B8)

Taking the n+1 values to be the unknown, while all the n present values are known, we end up with a system of simultaneous equations given as follows:

$$\begin{bmatrix} bl & \overline{A} & 0 & . & . & 0 \\ \overline{A} & \overline{B} & \overline{A} & . & . & 0 \\ 0 & \overline{A} & \overline{B} & \overline{A} & . & 0 \\ . & . & . & . & . & 0 \\ 0 & . & . & \overline{A} & \overline{B} & \overline{A} \\ 0 & 0 & . & . & \overline{A} & br \end{bmatrix} \begin{bmatrix} q_{n+1}^{1} \\ q_{n+1}^{2} \\ . \\ . \\ q_{n+1}^{m-1} \\ q_{n+1}^{m} \end{bmatrix} = \begin{bmatrix} d_{n}^{1} \\ d_{n}^{2} \\ . \\ . \\ d_{n}^{m-1} \\ d_{n}^{m} \end{bmatrix}$$

(B9)

where

$$d_{n}^{m} = \overline{C}q_{n}^{m+1} + \overline{D}q_{n}^{m} + \overline{C}q_{n}^{m-1}$$

(B10)

The bl and br are adjustable and are evaluated according to the boundary conditions. The above system is a tridiagonal system that is, except for three diagonals all the elements of the matrix in (A) are zero. Claerbout (1985), gives an elegant method for the solution of such systems. This method will be described in the following section.

Solving the Tridiagonal system of simultaneous equations.

Here we follow Claerbout (1985), to solve the system (B9). Let the simultaneous equations be written as a difference equation:

$$a_j q_{j+1} + b_j q_j + c_j q_{j-1} = d_j \qquad \text{(B11)}$$

with j = 1,....,m, where m is the number of traces in the gather.

Introduce two new variables e_j and f_j along with an equation

$$q_j = e_j q_{j+1} + f_j \qquad \text{(B12)}$$

which with shifted index can be written:

$$q_{j-1} = e_{j-1} q_j + f_{j-1} \qquad \text{(B13)}$$

Insert (B13) to (B11) :

$$a_j q_{j+1} + b_j q_j + c_j (e_{j-1} + q_j f_{j-1}) = d_j \qquad \text{(B14)}$$

Now rearrange (B14) to resemble (B12) :

$$q_j = \frac{-a_j}{b_j + c_j e_{j-1}} q_{j+1} + \frac{d_j - c_j f_{j-1}}{b_j + c_j e_{j-1}} \qquad \text{(B15)}$$

Compare (B15) with (B12) , we can see that recursion for e_j and f_j can be obtained as :

$$e_j = \frac{-a_j}{b_j + c_j e_{j-1}}$$

with $e_1 = -a_1 / b_1$

and

$$f_j = \frac{d_j - c_j f_{j-1}}{b_j + c_j e_{j-1}}$$

with $f_1 = d_1 / b_1$

For the right hand boundary we have from (B9) that:

$$A_{m-1}\, q_{m-1} + br\, q_m = d_m \qquad \text{(B16)}$$

and from (B13) replacing j by m we get:

$$q_{m-1} - e_{m-1}\, q_m = f_{m-1} \qquad \text{(B17)}$$

Solving the system of the two equations for q_m we have:

$$q_m = \frac{d_m - A_{m-1} f_{m-1}}{br + A_{m-1} e_{m-1}}$$

where br is evaluated using the 15 degree absorbing boundary condition, Clayton and Engquist, (1980).

Having e_j and f_j we can compute the rest of q_j 's from equation (B13) going backwards.

APPENDIX C

F-D FORMULATION OF THE 15^0 3-D PARAXIAL WAVE EQUATION

The monochromatic wavefield at Q(x,y,z) is calculated by discretizing the differential equation (4.15), and approximating Q(x,y,z) by a mesh function $q^m_{n,l}$ where

n=i Δx , l=iΔy, with i=1,.....,Nx and m=k Δz , with k=1,....,Nz.

where Δx, Δy and Δz are the finite difference mesh grid spacings for the horizontal and vertical axes respectively. Nx indicates the number of traces in the gather and Nz the number of depth steps.

Replacing the differentials of equation (4.15) by their respective finite differences and changing the sign of the z differentials to evaluate upgoing waves one has:

$$- 2i\mu\delta_z Q + \delta_{xx} Q + \delta_{yy} Q = 0 \qquad \text{(C1)}$$

where $\delta_z = \partial/\partial z$, $\delta_{xx} = \partial^2 / \partial x^2$, $\delta_{yy} = \partial^2 / \partial y^2$, and $\mu(x,z)=\omega/v(x,z)$.

A higher order (fourth) of accuracy for the approximation of the second order differential is the Douglas formula and is given by (Mitchell and Griffiths, 1980; Claerbout, 1985) :

$$\delta_{xx} = \frac{1}{\Delta x^2}\left(\frac{\delta^2}{1+\gamma\delta^2}\right) \qquad \text{(C2)}$$

where Δx is the grid spacing and $\gamma = 1/12$ (Mitchell and Griffiths (1980) or 1/6 Claerbout (1985), and δ^2 is the operator:

$$\begin{bmatrix} & 1 & \\ 1 & -4 & 1 \\ & 1 & \end{bmatrix}$$

An implicit Crank -Nicolson (Mitchell and Griffiths, 1980; Claerbout 1976,1985) finite difference algorithm scheme is employed to solve the resultant difference equation. This algorithm is unconditionally stable (Mitchell and Griffiths, 1980) and involves more than one point at the advanced depth level according to :

$$q^m_{n,l} = \frac{1}{2}\left(q^{m+1}_{n,l} + q^m_{n,l} \right) \qquad \text{(C3)}$$

The finite difference approximation of the differential equation (C1) employing equations (C2) and (C3) reads:

$$-\frac{2\mu i}{\Delta z}\left(q^{m+1}_{n,l} - q^m_{n,l}\right) + \frac{1}{2}\frac{\delta^2}{\Delta x^2(1+\gamma\delta^2)} q^{m+1}_{n,l} + \frac{1}{2}\frac{\delta^2}{\Delta x^2(1+\gamma\delta^2)} q^m_{n,l}$$
$$+ \frac{1}{2}\frac{\delta^2}{\Delta y^2(1+\gamma\delta^2)} q^{m+1}_{n,l} + \frac{1}{2}\frac{\delta^2}{\Delta y^2(1+\gamma\delta^2)} q^m_{n,l} = 0 \qquad \text{(C4)}$$

Assuming that $\Delta x = \Delta y$ and multiply throughout by $\Delta x^2 (1+\gamma\delta^2)$ we have:

$$-\frac{2\mu i \Delta x^2}{\Delta z}(1+\gamma\delta^2)\left(q^{m+1}_{n,l} - q^m_{n,l}\right) + \frac{2\delta^2}{2}\left(q^m_{n+1} + q^m_n\right) = 0 \qquad \text{(C5)}$$

Simplifying by setting the laterally varying coefficients as A and B one has:

A=(1.0,0.)

$$B = \frac{2\mu i \Delta x^2}{\Delta z} -$$

and apply the operator δ^2 we have:

$$- B\, q_{n,l}^{m+1} + (A-\gamma B)\left[q_{n-1,l}^{m+1} + q_{n+1,l}^{m+1} - 4\, q_{n,l}^{m+1} + q_{n,l+1}^{m+1} + q_{n,l-1}^{m+1}\right] =$$
$$- B\, q_{n,l}^{m} + (-A-\gamma B)\left[q_{n-1,l}^{m} + q_{n+1,l}^{m} - 4\, q_{n,l}^{m} + q_{n,l+1}^{m} + q_{n,l-1}^{m}\right] \qquad (C6)$$

Separate the terms with index m+1 to the left hand side and the terms with index n to the right hand side and after some algebraic simplifications upon replacing with:

$$\overline{A} = (A - \gamma B)$$
$$\overline{B} = (-4A - B + 4\gamma B)$$

and for the right hand side:

$$\overline{C} = (-A - \gamma B)$$
$$\overline{D} = (4A - B + 4\gamma B)$$

or in compact form:

$$-\overline{A}q_{n+1,l}^{m+1} - \overline{A}q_{n,l+1}^{m+1} + \overline{B}q_{n,l}^{m+1} - \overline{A}q_{n-1,l}^{m+1} - \overline{A}q_{n,l-1}^{m+1} = -\overline{C}q_{n+1,l}^{m} - \overline{C}q_{n,l+1}^{m} + \overline{D}q_{n,l}^{m} - \overline{C}q_{n-1,l}^{m} - \overline{C}q_{n,l-1}^{m} \qquad (C8)$$

Taking the m+1 values to be the unknown, while all the n present values are known, we end up with a system of simultaneous equations in pentadiagonal form which can be solved using under-relaxation as described in Chapter 4.

APPENDIX D

D1. Stability analysis for the 65 degrees paraxial approximation.

The problem of stability of a finite difference calculation used to solve equation (3.18) consists of finding a condition under which the difference:

$$q_n^m - \overline{q_n^m} = (Z_n^m) \qquad \text{(D1)}$$

between the theoretical and numerical solutions of the difference equation bounded as n tends to infinity, (Mitchell and Griffiths, 1980).

The von Neumann procedure consists of considering an harmonic decomposition of the error $Z_n{}^m$ at grid points at a given depth level leading to error function

$$E(x) = \sum_j A_j e^{i b_j x} \qquad \text{(D2)}$$

where in general the frequencies $| b_j |$ and j are arbitrary.

It is necessary to consider only the single term e^{ibx} where b is a any real number. For convenience, suppose that the depth being considered is z=0. To investigate the error propagation as z increases, it is necessary to find a solution of the finite difference equation (B1) which reduces to e^{ibx} where z= 0. Let such a solution be:

$e^{az} e^{ibx}$ where a = a(b) is complex in general. The original error component will not grow with increasing depth if $| e^{ak} | \leq 1$ for all a.

Boundary conditions are neglected in the von Neumann method (Mitchell and Griffiths, 1980), which applies in theory only to pure initial value problems. It does however provide necessary conditions for stability of constant coefficient problems

regardless of the type of boundary conditions. Substitute

$$Z_n^m = e^{a n k}\, e^{i b m h}$$

where $k = \Delta z$ and $h = \Delta x$, into the difference equation (B6) with A=-A and B = -B indicating downgoing waves and divide by $e^{a n k}\, e^{i b m h}$. We have:

$$\begin{aligned} &A e^{ak} e^{ibh} - 2A e^{ak} + A e^{ak} e^{-ibh} \\ &- A e^{ibh} + 2A - A e^{-ibh} + \tfrac{1}{2} e^{ak} e^{ibh} - e^{ak} + \tfrac{1}{2} e^{ak} e^{-ibh} + \\ &+ \tfrac{1}{2} e^{ibh} - 1 + \tfrac{1}{2} e^{-ibh} + B e^{ak} - B + B\gamma e^{ak} e^{ibh} - 2B\gamma e^{ak} + B\gamma e^{ak} e^{-ibh} - \\ &- B\gamma e^{ibh} + 2B\gamma + B\gamma e^{-ibh} = 0 \end{aligned} \qquad \text{(D3)}$$

Multiplying by e^{ak} throughout and recognizing that exponential terms can be expressed as cosines we have:

$$\begin{aligned} &A e^{2ak}(2\cos(bh) - 2) - A e^{ak}(2\cos(bh) - 2) + \tfrac{1}{2} e^{2ak}(2\cos(bh) - 2) + \tfrac{1}{2} e^{ak}(2\cos(bh) - 2) \\ &+ B e^{2ak} - B e^{ak} + B\gamma e^{2ak}(2\cos(bh) - 2) - B\gamma e^{ak}(2\cos(bh) - 2) = 0 \end{aligned} \qquad \text{(D4)}$$

Recognizing that $(1 - \cos(t)) = 2\sin^2(t/2)$ and simplifying by denoting $\xi = e^{ak}$ and multiplying by -1 throughout we have:

$$\begin{aligned} &4A\xi^2 \sin^2(bh/2) - 4A\xi \sin^2(bh/2) + 2\xi^2 \sin^2(bh/2) + \\ &+ 2\xi \sin^2(bh/2) - B\xi^2 + B\xi + 4B\gamma\xi^2 \sin^2(bh/2) - 4B\gamma\xi \sin^2(bh/2) = 0 \end{aligned} \qquad \text{(D5)}$$

Factoring out ξ^2 and ξ we get a second order polynomial equation in ξ as:

$$\xi^2(4A \sin^2(bh/2) + 2\sin^2(bh/2) - B + 4B\gamma \sin^2(bh/2))$$

$$+\xi(-4\ A\ \sin^2(b\,h/2)+ 2\ \sin^2(b\,h/2)+ B - 4\ B\ \gamma\sin^2(b\,h/2)) = 0 \qquad (D6)$$

Setting $S = \sin^2(b\,h/2)$ and recognizing that the coefficients of ξ^2 and ξ are the same except the -2 S and +2 S terms we get a simplified equation:

$$\xi^2 (C + 2S) - \xi (C - 2S) = 0 \qquad (D7)$$

Solving for ξ we get the two solutions $\xi_1 = 0$ and $\xi_2 = (C-2S)/(C+2S)$

Since C is a pure imaginary number the amplification factor - ξ then takes the form of a complex number divided by its complex conjugate. Expressing the complex number in polar form it becomes clear that such a number has a magnitude of unity. Hence, $|\xi| < 1$ and this satisfies the unconditional stability criterion.

D2 Absorbing Boundary conditions

We have used the so called 15 degree paraxial absorbing boundary conditions in the solution of equation (3.18) based on Clayton and Enquist (1980). The effectiveness of these conditions can be shown by comparing their effective reflection coefficients at the boundaries. The 2-D scalar wave equation is given by:

$$P_{xx} + P_{zz} = \frac{1}{v^2} P_{tt} \qquad \text{(D8)}$$

Taking the 15 degree paraxial approximation we have:

$$P_x + \frac{1}{v} P_t = 0 \qquad \text{(D9)}$$

Consider an incident plane wave travelling at the -z, +x direction according to

$$P_i = e^{(ik_x x - ik_z z - i\omega t)} \qquad \text{(D10)}$$

which initiates a reflection from the right boundary of the form

$$P_r = re^{(-ik_x x - ik_z z - i\omega t)} \qquad \text{(D11)}$$

where r is the effective reflection coefficient. Locally near the boundary the wave field $(P_i + P_r)$ will satisfy the boundary condition given above thus obtaining:

$$ik_x - irk_x - i\frac{\omega}{v} - ir\frac{\omega}{v} = 0 \qquad \text{(D12)}$$

or

$$r = \frac{k_x - \frac{\omega}{v}}{k_x + \frac{\omega}{v}} \qquad \text{(D13)}$$

Recognizing that $k_x = k \sin(\theta)$ where $k = \omega/v$ and θ is the angle between the travelling

wave and the vertical side boundary we get:

$$r = \frac{\sin(\theta) - 1}{\sin(\theta) + 1} \qquad \text{(D14)}$$

So when the wave hits the boundary with θ=0 degrees then r=-1. These boundary conditions are match with the exact paraxial wave equation (Clayton and Enquist, 1980). Similarly, when the wave hits the boundary with greater angles then we can see that the effective reflection coefficient decreases.

APPENDIX E

E1. Stability analysis for the 45 degrees 3-D paraxial approximation.

The problem of stability of a finite difference calculation used to solve equation (6.15) consists of finding a condition under which the difference:

$$q^m_{n,l} - \overline{q^m_{n,l}} = (Z^m_{n,l}) \qquad \text{(E1)}$$

between the theoretical and numerical solutions of the difference equation bounded as n tends to infinity, (Mitchell and Griffiths, 1980) in a similar manner as we did for the 2-D case.

The von Neumann procedure consists of considering a harmonic decomposition of the error $Z_{n,l}{}^m$ at grid points at a given depth level leading to error function

$$E(x) = \sum_j A_j e^{i\, b_j\, x} \qquad \text{(E2)}$$

where in general the frequencies $| b_j |$ and *j* are arbitrary.

It is necessary to consider only the single term e^{ibx} where b is a any real number. For convenience, suppose that the depth being considered is z=0. To investigate the error propagation as z increases, it is necessary to find a solution of the finite difference equation (6.17) which reduces to $e^{ibx}\, e^{iby}$ where z= 0. Let such a solution be:

$e^{az}\, e^{ibx}\, e^{iby}$ where a = a(b) is complex in general. The original error component will not grow with increasing depth if $| e^{ak} | \leq 1$ for all a.

Substitute

$$Z^{m}_{n,l} = e^{amk}\ e^{ibnh}\ e^{iblh}$$

where k = Δz and h = Δx= Δy, into the difference equation (6.17) with A=-A and B = -B and C = -C indicating downgoing waves and divide by $e^{amk}\ e^{ibnh} e^{iblh}$. We have:

$$e^{ak}\left\{\begin{array}{c}\left(B - 8\gamma B - 4A - 4 + 4\gamma^2 B + 4\gamma A + 4\gamma - 4C\right) + \left(2e^{-ibh} + 2e^{ibh}\right) \\ \left(2\gamma B + A + 1 - 2\gamma^2 B - 2\gamma A - 2\gamma + 2C\right) + \left(e^{-2ibh} + 2 + e^{2ibh}\right)\left(\gamma^2 B + \gamma A - \gamma - C\right)\end{array}\right\} =$$
$$\left\{\begin{array}{c}\left(B - 8\gamma B - 4A + 4 + 4\gamma^2 B + 4\gamma A - 4\gamma + 4C\right) + \left(2e^{-ibh} + 2e^{ibh}\right) \\ \left(2\gamma B + A - 1 - 2\gamma^2 B - 2\gamma A + 2\gamma - 2C\right) + \left(e^{-2ibh} + 2 + e^{2ibh}\right)\left(\gamma^2 B + \gamma A - \gamma + C\right)\end{array}\right\}$$

(E3)

Multiplying by e^{ak} throughout and simplifying by denoting $\xi = e^{ak}$ we have:

$$\xi^2 B + 8\gamma B\xi^2[-1+\cos(bh)] + 4A\xi^2[-1+\cos(bh)] + 4\xi^2[-1+\cos(bh)] + 6\gamma^2 B\xi^2[1-\cos(bh)]$$
$$+ 6\gamma A\xi^2[1-\cos(bh)] + 6\gamma\xi^2[1-\cos(bh)] + 6C\xi^2[-1+\cos(bh)] =$$
$$\xi B + 8\gamma B\xi[-1+\cos(bh)] + 4A\xi[-1+\cos(bh)] + 4\xi[1-\cos(bh)] + 6\gamma^2 B\xi[1-\cos(bh)]$$
$$+ 6\gamma A\xi[1-\cos(bh)] + 6\gamma\xi[-1+\cos(bh)] - 6C\xi[1-\cos(bh)]$$

(E4)

Recognizing that $(1-\cos(t)) = 2\sin^2(t/2)$ we have:

$$\xi^2 B - 16\gamma B\xi^2\left[\sin^2\left(\frac{bh}{2}\right)\right] - 8A\xi^2\left[\sin^2\left(\frac{bh}{2}\right)\right] - 8\xi^2\left[\sin^2\left(\frac{bh}{2}\right)\right] + 12\gamma^2 B\xi^2\left[\sin^2\left(\frac{bh}{2}\right)\right]$$
$$+ 12\gamma A\xi^2\left[\sin^2\left(\frac{bh}{2}\right)\right] + 12\gamma\xi^2\left[\sin^2\left(\frac{bh}{2}\right)\right] - 12C\xi^2\left[\sin^2\left(\frac{bh}{2}\right)\right]$$
$$-\xi B + 16\gamma B\xi\left[\sin^2\left(\frac{bh}{2}\right)\right] + 8A\xi\left[\sin^2\left(\frac{bh}{2}\right)\right] - 8\xi\left[\sin^2\left(\frac{bh}{2}\right)\right] - 12\gamma^2 B\xi\left[\sin^2\left(\frac{bh}{2}\right)\right]$$
$$- 12\gamma A\xi\left[\sin^2\left(\frac{bh}{2}\right)\right] + 12\gamma\xi\left[\sin^2\left(\frac{bh}{2}\right)\right] + 12C\xi\left[\sin^2\left(\frac{bh}{2}\right)\right] = 0$$

(E5)

Factoring out ξ^2 and ξ we get a second order polynomial equation in ξ as:

$$\xi^2 \begin{Bmatrix} B - 16\gamma B\left[\sin^2\left(\frac{bh}{2}\right)\right] - 8A\left[\sin^2\left(\frac{bh}{2}\right)\right] - 8\left[\sin^2\left(\frac{bh}{2}\right)\right] \\ +12\gamma^2 B\left[\sin^2\left(\frac{bh}{2}\right)\right] + 12\gamma A\left[\sin^2\left(\frac{bh}{2}\right)\right] + 12\gamma\left[\sin^2\left(\frac{bh}{2}\right)\right] - 12C\left[\sin^2\left(\frac{bh}{2}\right)\right] \end{Bmatrix}$$

$$\xi \begin{Bmatrix} -B + 16\gamma B\left[\sin^2\left(\frac{bh}{2}\right)\right] + 8A\left[\sin^2\left(\frac{bh}{2}\right)\right] - 8\left[\sin^2\left(\frac{bh}{2}\right)\right] \\ -12\gamma^2 B\left[\sin^2\left(\frac{bh}{2}\right)\right] - 12\gamma A\left[\sin^2\left(\frac{bh}{2}\right)\right] + 12\gamma\left[\sin^2\left(\frac{bh}{2}\right)\right] + 12C\left[\sin^2\left(\frac{bh}{2}\right)\right] \end{Bmatrix} = 0 \qquad \text{(E6)}$$

Setting $S = \sin^2 (b\,h/\,2)$ and recognizing that the coefficients of ξ^2 and ξ are the same except the -8 S and + 12γ S terms we get a simplified equation with $S' = 12\gamma S - 8S$:

$$\xi^2 (C + S') - \xi (C - S') = 0 \qquad \text{(E7)}$$

Solving for ξ we get the two solutions

$\xi_1 = 0$ and

$$\xi_2 = \frac{(C - S')}{(C + S')} \qquad \text{(E8)}$$

Since C is a pure imaginary number the amplification factor - ξ then takes the form of a complex number divided by its complex conjugate. Expressing the complex number in polar form it becomes clear that such a number has a magnitude of unity. Hence, $|\xi| < 1$ and this satisfies the unconditional stability criterion.

E2 Absorbing Boundary conditions

I used the 15 degree paraxial absorbing boundary conditions in the solution of equation (6.15) based on Clayton and Enquist (1980). The method is similar to that shown in Appendix D. The 3-D scalar wave equation is given by:

$$P_{xx} + P_{yy} + P_{zz} = \frac{1}{v^2} P_{tt} \tag{E9}$$

Taking the 15 degree paraxial approximation we have:

$$P_x + P_y + \frac{1}{v} P_t = 0 \tag{E10}$$

Consider an incident plane wave travelling at the -z, +x and +y direction according to

$$P_i = e^{(ik_x x + ik_x x - ik_z z - i\omega t)} \tag{E11}$$

which initiates a reflection from the right boundary of the form

$$P_r = re^{(-ik_x x - ik_x x - ik_z z - i\omega t)} \tag{E12}$$

where r is the effective reflection coefficient. Locally near the boundary the wave field $(P_i + P_r)$ will satisfy the boundary condition given above thus obtaining:

$$ik_x + ik_y - irk_x - irk_y - i\frac{\omega}{v} - ir\frac{\omega}{v} = 0 \tag{E13}$$

or

$$r = \frac{k_x + k_y - \frac{\omega}{v}}{k_x + k_y + \frac{\omega}{v}} \tag{E14}$$

recognizing that $k_x = k \sin\theta$ and $k_y = k\cos\theta$ where $k = \omega/v$ and θ is the angle between the travelling wave and the vertical side boundary we get:

$$r = \frac{\sin\theta + \cos\theta - 1}{\sin\theta + \cos\theta + 1} \qquad \text{(E15)}$$

When the wave hits the boundary with $\theta=0$ and 90 degrees .hen r=0. Similarly, when the wave hits the boundary with greater angles between 0 and 90 degrees then we can see that the effective reflection coefficient increases to a maximum of r=0.16.